Robust and optimal multi-iterative techniques for IgA Galerkin linear systems

Marco Donatelli *Carlo Garoni* *Carla Manni*
Stefano Serra-Capizzano *Hendrik Speleers*

Report TW 643, March 2014

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

Robust and optimal multi-iterative techniques for IgA Galerkin linear systems

Marco Donatelli *Carlo Garoni* *Carla Manni*
Stefano Serra-Capizzano *Hendrik Speleers*

Report TW 643, March 2014

Department of Computer Science, KU Leuven

Abstract

We consider fast solution methods for large linear systems arising from the Galerkin approximation based on B-splines of classical d-dimensional elliptic problems, $d \geq 1$, in the context of isogeometric analysis. Our ultimate goal is to design iterative algorithms with the following two properties. First, their computational cost is optimal, that is linear with respect to the number of degrees of freedom. Second, they are totally robust, i.e., their convergence speed is substantially independent of all the relevant parameters: in our case, this means the matrix size (related to the finesse parameter), the spline degree (associated to the approximation order), and the dimensionality d of the problem. We review several methods like PCG, multigrid, multi-iterative algorithms, and we carefully show how their numerical practical behavior (in terms of convergence speed) can be completely understood through the notion of spectral distribution, i.e., through a compact symbol which describes the global eigenvalue behavior of the related stiffness matrices. As a final step, we show how we can design an optimal and totally robust multi-iterative method, by taking into account the analytic features of the symbol. A wide variety of numerical experiments, few open problems, and perspectives are presented and critically discussed.

Keywords : Isogeometric analysis; Galerkin method; B-splines; Toeplitz matrices; symbol; PCG, multigrid, and multi-iterative methods.

Robust and optimal multi-iterative techniques for IgA Galerkin linear systems

Marco Donatelli[a], Carlo Garoni[a], Carla Manni[b],
Stefano Serra-Capizzano[a], Hendrik Speleers[c]

[a] *Department of Science and High Technology, University of Insubria, Via Valleggio 11, 22100 Como, Italy*
[b] *Department of Mathematics, University of Rome 'Tor Vergata', Via della Ricerca Scientifica, 00133 Rome, Italy*
[c] *Department of Computer Science, University of Leuven, Celestijnenlaan 200A, 3001 Heverlee (Leuven)*

Abstract

We consider fast solution methods for large linear systems arising from the Galerkin approximation based on B-splines of classical d-dimensional elliptic problems, $d \geq 1$, in the context of isogeometric analysis. Our ultimate goal is to design iterative algorithms with the following two properties. First, their computational cost is optimal, that is linear with respect to the number of degrees of freedom. Second, they are totally robust, i.e., their convergence speed is substantially independent of all the relevant parameters: in our case, this means the matrix size (related to the finesse parameter), the spline degree (associated to the approximation order), and the dimensionality d of the problem. We review several methods like PCG, multigrid, multi-iterative algorithms, and we carefully show how their numerical practical behavior (in terms of convergence speed) can be completely understood through the notion of spectral distribution, i.e., through a compact symbol which describes the global eigenvalue behavior of the related stiffness matrices. As a final step, we show how we can design an optimal and totally robust multi-iterative method, by taking into account the analytic features of the symbol. A wide variety of numerical experiments, few open problems, and perspectives are presented and critically discussed.

Keywords: Isogeometric analysis; Galerkin method; B-splines; Toeplitz matrices; symbol; PCG, multigrid, and multi-iterative methods.

1. Introduction

In this paper we design and analyze fast solvers for the large linear systems resulting from the Galerkin method based on B-splines in the context of Isogeometric Analysis (IgA) [13, 28], applied to diffusion dominated elliptic Partial Differential Equations (PDEs) on the d-dimensional cube $(0,1)^d$ as in equation (2.1). In a recent work [25], the spectral properties of the related stiffness matrices have been studied in some detail. In particular, the spectral localization and the conditioning were investigated, while the asymptotic spectral distribution (as the matrix size tends to infinity) has been compactly characterized in terms of a d-variate

Email addresses: marco.donatelli@uninsubria.it (Marco Donatelli), carlo.garoni@uninsubria.it (Carlo Garoni), manni@mat.uniroma2.it (Carla Manni), stefano.serrac@uninsubria.it (Stefano Serra-Capizzano), hendrik.speleers@cs.kuleuven.be (Hendrik Speleers)

trigonometric polynomial, denoted by f_p in the one-dimensional case ($d = 1$) and by $f_{p_1,p_2}^{(\nu_1,\nu_2)}$ in the two-dimensional case ($d = 2$). Here, the parameters p, p_1, p_2 are indices of the specific IgA approximation, namely the degrees of the used B-splines (for the 2D case we have two spline degrees, p_1 for the x-direction and p_2 for the y-direction). In all the considered cases, in analogy with Finite Difference (FD) and Finite Element (FE) cases, the conditioning grows as $m^{2/d}$, where m is the matrix size, d is the dimensionality of the elliptic problem, and 2 is the order of the elliptic operator. As expected, the approximation parameters p, p_1, p_2 play a limited role, because they only characterize the constant in the expression $O(m^{2/d})$. The growth of the condition number implies that all classical stationary iterative methods and the Krylov methods are not optimal, in the sense that the number of iterations for reaching a preassigned accuracy ϵ is a function diverging to infinity, as the matrix size m tends to infinity. In order to bypass this difficulty, we consider Preconditioned Conjugate Gradient (PCG) methods, multigrid techniques (V- and W-cycles), and multi-iterative algorithms, with the aim of designing optimal (and totally robust) iterative solvers.

We specify that the notion of optimality for an iterative method is twofold. First, the number of iterations for reaching a preassigned accuracy ϵ must be bounded by a constant $c(\epsilon)$ independent of the matrix size. This is also known as the *optimal convergence rate condition*. For stationary iterative methods, it translates into the requirement that the spectral radius of the iteration matrix is bounded by a constant $c < 1$ independent of the matrix size. Second, when the matrix size goes to infinity, the cost per iteration must be asymptotically of the same order as the cost of multiplying the matrix by a vector. In this paper, we are not concerned with the second requirement which is 'for free' in the considered iterative solvers, given the banded-ness of the involved matrices, and we focus our attention on the first one, the 'optimal convergence rate condition', as a simplified definition of optimality.

In analogy with the FD/FE setting (we refer to [4, 35, 36]), in this paper we exploit the spectral information represented by the symbol f_p or $f_{p_1,p_2}^{(\nu_1,\nu_2)}$ for devising optimal and totally robust multi-iterative techniques based on multigrid and PCG methods. In particular, we follow (see [14, 20, 34]) a sort of 'canonical procedure' for creating – inspired by the symbol – PCG, two-grid, V-cycle, W-cycle methods from which we expect optimal convergence properties, at least in certain subspaces or with respect to some of the parameters (we refer to [16] for rigorous proofs of convergence).

For all the considered basic methods, the convergence rate is not optimal or is not satisfactory at least for large p, p_1, p_2 (the approximation parameters). For example, in the case of multigrid methods we have theoretical optimality, but the convergence rates are close to 1, when the approximation parameters increase. Indeed, the spectral radius of the multigrid iteration matrices tends to 1 exponentially as p increases. This catastrophic behavior is due to the analytical properties of the symbols $f_p, f_{p_1,p_2}^{(\nu_1,\nu_2)}$ and can be understood in terms of the theory of Toeplitz matrices, because $f_p(\pi), f_{p_1,p_2}^{(\nu_1,\nu_2)}(\theta_1,\pi), f_{p_1,p_2}^{(\nu_1,\nu_2)}(\pi,\theta_2)$ are positive but converge to zero exponentially fast, with respect to the degrees of the B-splines. The considered Galerkin matrices are ill-conditioned in subspaces of low frequencies (dictated by the relations $f_p(0) = f_{p_1,p_2}^{(\nu_1,\nu_2)}(0,0) = 0$), but this is a common property shared by all the matrices obtained via local methods such as e.g. FDs/FEs. In addition, for large p, p_1, p_2, our matrices possess a non-canonical behavior in a high frequency subspace, due to the fact that $f_p(\pi), f_{p_1,p_2}^{(\nu_1,\nu_2)}(\theta_1,\pi), f_{p_1,p_2}^{(\nu_1,\nu_2)}(\pi,\theta_2)$ are numerically zero. We refer to [16] for a theoretical (and rather technical) proof of these facts.

We follow two directions to address the above intrinsic difficulty. First, we change the size reduction strategy as suggested in [18], thanks to the notion of g-Toeplitz (see [29, 39, 40]). Second, we enrich our multigrid procedures by varying the choice of the smoothers, in the sense of the multi-iterative idea [30]. The second idea turns out to be more successful than

the first one. In fact, the best techniques, those showing at the same time optimality and total robustness, fall in the class of multi-iterative methods involving a few PCG smoothing steps at the finest level, where the related preconditioner is chosen as the Toeplitz matrix generated by a specific function coming from the factorization of the symbol. In such a way, the preconditioner works in the subspace of high frequencies where there exists the ill-conditioning induced by the parameters p, p_1, p_2 (and which becomes worse when increasing the dimensionality d), while the standard choice of the prolongation and restriction operators as in [20, 34] is able to cope with the standard ill-conditioning in the low frequency subspace. The combination of the two complementary spectral behaviors induces a global method which is fast in any frequency domain, independently of all the relevant parameters.

Regarding fast solvers for IgA linear systems, we observe that the literature on this concern seems to be very recent and quite limited:

1. (geometric) multigrid methods [22],
2. algebraic multilevel preconditioners [23],
3. BPX (Bramble-Pasciak-Xu) preconditioners [12],
4. BDDC (Balancing Domain Decomposition by Constraints) preconditioners [5],
5. Schwarz preconditioners [6].

It is worthwhile noticing that the basic ingredients of the techniques used so far is not different from our approach: different kinds of preconditioning and various types of multigrid algorithms. However, the innovative aspect of our approach is that the choice of the ingredients of the global solver (in fact a multi-iterative solver) is guided by the knowledge of the symbol, which in turns offers an approximate understanding of the subspaces where the stiffness matrix is ill-conditioned.

Although in some of the contributions (see e.g. [22, 23]) the bad dependency on the parameters p, p_1, p_2 was observed, one did not realize that spurious small eigenvalues are present already for $p, p_1, p_2 \geq 4$ and that the related eigenspace largely intersects the high frequencies, see Section 4 for the details. The latter phenomenon is indeed unexpected in this context, since high frequency eigenspaces related to small eigenvalues are typical of matrices resulting from the approximation of integral operators, like in the setting of blurring models in imaging and signal processing (see e.g. [17, 19]).

By exploiting the information from the symbol, we are able to design a cheap (indeed optimal) solver of multi-iterative type, whose convergence speed is independent of all the relevant parameters of the problems: namely the finesse parameter (related to the size of the matrices), the approximation parameters p, p_1, p_2 and the dimensionality d. This is clearly illustrated by our best algorithmic proposals in Sections 6 and 7. We remark that the numerical experiments are carried out for $d = 1, 2$, but the proposal is uniformly applicable for every $d \geq 1$ and its generality easily follows from the expression of the symbol in d dimensions (compare the analysis in Section 4.2 and Section 4.3).

The paper is organized as follows. In Section 2 we introduce the considered model problem and the related matrices. Section 3 is devoted to the notion of symbol; it considers the noteworthy example of d-level Toeplitz matrices, and it provides a practical guide on how to use the symbol for sequences of matrices, especially of Toeplitz type. Section 4 studies the symbols $f_p, f_{p_1,p_2}^{(\nu_1,\nu_2)}$ which are related to 1D and 2D stiffness matrices approximating the given differential problems. Moreover, the general case in d dimensions is briefly sketched in Section 4.3. Section 5 addresses the multi-iterative idea, and lists our algorithms, by stressing how the prolongation/restriction operators and the preconditioners are chosen by following the spectral information. Section 6 and Section 7 present and discuss numerical experiments for the two-grid methods in 1D and 2D, respectively. Section 8 contains numerical results

regarding the V-cycle and W-cycle multigrid algorithm, and a brief discussion on the role of the advection term. We conclude the work in Section 9, by emphasizing perspectives and open problems: among others, the generalization of the present approach to complex geometries, variable coefficients, and the study of the effects of a dominating advection term.

2. The d-dimensional problem setting

Our model problem is the following elliptic problem:

$$\begin{cases} -\Delta u + \boldsymbol{\beta} \cdot \nabla u + \gamma u = \mathrm{f}, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases} \tag{2.1}$$

with $\Omega = (0,1)^d$, $\mathrm{f} \in L^2(\Omega)$, $\boldsymbol{\beta} = (\beta_1, \dots, \beta_d) \in \mathbb{R}^d$, $\gamma \geq 0$. The weak form of (2.1) reads as follows: find $u \in H_0^1(\Omega)$ such that

$$a(u, v) = \mathrm{F}(v), \qquad \forall v \in H_0^1(\Omega), \tag{2.2}$$

where $a(u,v) = \int_\Omega (\nabla u \cdot \nabla v + \boldsymbol{\beta} \cdot \nabla u \, v + \gamma u v) \, \mathrm{d}\Omega$ and $\mathrm{F}(v) = \int_\Omega \mathrm{f} v \, \mathrm{d}\Omega$. It is known, see e.g. [11], that there exists a unique solution u of (2.2), the so-called weak solution of (2.1).

In the Galerkin method, we look for an approximation $u_{\mathcal{W}}$ of u by choosing a finite dimensional approximation space $\mathcal{W} \subset H_0^1(\Omega)$ and by solving the following (Galerkin) problem: find $u_{\mathcal{W}} \in \mathcal{W}$ such that

$$a(u_{\mathcal{W}}, v) = \mathrm{F}(v), \qquad \forall v \in \mathcal{W}. \tag{2.3}$$

Let $\dim \mathcal{W} = m$, and fix a basis $\{\varphi_1, \dots, \varphi_m\}$ for $\mathcal{W}$. It is known that the problem (2.3) always has a unique solution $u_{\mathcal{W}} \in \mathcal{W}$, which can be written as $u_{\mathcal{W}} = \sum_{j=1}^m u_j \varphi_j$ and can be computed as follows: find $\mathbf{u} = (u_1, \dots, u_m)^T \in \mathbb{R}^m$ such that

$$A\mathbf{u} = \mathbf{b}, \tag{2.4}$$

where $A := [a(\varphi_j, \varphi_i)]_{i,j=1}^m \in \mathbb{R}^{m \times m}$ is the stiffness matrix, and $\mathbf{b} := [\mathrm{F}(\varphi_i)]_{i=1}^m \in \mathbb{R}^m$. The matrix A is positive definite in the sense that $\mathbf{v}^T A \mathbf{v} > 0$, $\forall \mathbf{v} \in \mathbb{R}^m \setminus \{\mathbf{0}\}$.

In classical FE methods the approximation space $\mathcal{W}$ is usually a space of C^0 piecewise polynomials vanishing on $\partial\Omega$, whereas in the IgA framework $\mathcal{W}$ is a spline space with higher continuity, see [13, 28].

Now we focus on our model problem for $d = 1$:

$$\begin{cases} -u'' + \beta u' + \gamma u = \mathrm{f}, & \text{in } (0,1), \\ u(0) = 0, \quad u(1) = 0, \end{cases} \tag{2.5}$$

with $\mathrm{f} \in L^2(0,1)$, $\beta \in \mathbb{R}$, $\gamma \geq 0$.

In the IgA framework based on B-splines, we approximate the (weak) solution u of (2.5) with the approximation space $\mathcal{W}$ chosen as a space of smooth polynomial splines and with the basis chosen as the B-spline basis. More precisely, for $p \geq 1$ and $n \geq 2$, let

$$\begin{aligned} \mathcal{V}_n^{[p]} &:= \left\{ s \in C^{p-1}[0,1] : \, s_{|\left[\frac{i}{n}, \frac{i+1}{n}\right)} \in \mathbb{P}_p, \quad \forall i = 0, \dots, n-1 \right\}, \\ \mathcal{W}_n^{[p]} &:= \{ s \in \mathcal{V}_n^{[p]} : \, s(0) = s(1) = 0 \} \subset H_0^1(0,1), \end{aligned}$$

where $\mathbb{P}_p$ stands for the space of polynomials of degree less than or equal to p. It is known that $\dim \mathcal{V}_n^{[p]} = n + p$ and $\dim \mathcal{W}_n^{[p]} = n + p - 2$. Then, we choose $\mathcal{W} = \mathcal{W}_n^{[p]}$, for some $p \geq 1$, $n \geq 2$, and for $\mathcal{W}_n^{[p]}$ we choose the B-spline basis $\{N_{2,[p]}, \dots, N_{n+p-1,[p]}\}$ described in [25, Section 4] (see also [8]).

Definition 2.1. *Given the knot sequence*

$$t_1 = \cdots = t_{p+1} = 0 < t_{p+2} < \cdots < t_{p+n} < 1 = t_{p+n+1} = \cdots = t_{2p+n+1},$$

with

$$t_{p+i+1} := \frac{i}{n}, \quad \forall i = 0, \ldots, n,$$

the B-splines $N_{i,[k]} : [0,1] \to \mathbb{R}$ of degree k are defined (recursively) on this knot sequence as follows: for every (k,i) such that $0 \le k \le p$, $1 \le i \le (n+p)+p-k$,

$$N_{i,[0]}(x) := \begin{cases} 1, & \text{if } x \in [t_i, t_{i+1}), \\ 0, & \text{elsewhere,} \end{cases}$$

and

$$N_{i,[k]}(x) := \frac{x - t_i}{t_{i+k} - t_i} N_{i,[k-1]}(x) + \frac{t_{i+k+1} - x}{t_{i+k+1} - t_{i+1}} N_{i+1,[k-1]}(x), \quad k > 0,$$

where we assume that a fraction with zero denominator is zero.

When choosing the B-spline basis of degree p in Definition 2.1, the corresponding Galerkin stiffness matrix is given by

$$A_n^{[p]} := \left[a(N_{j+1,[p]}, N_{i+1,[p]})\right]_{i,j=1}^{n+p-2} = nK_n^{[p]} + \beta H_n^{[p]} + \frac{\gamma}{n} M_n^{[p]} \in \mathbb{R}^{(n+p-2)\times(n+p-2)},$$

where

$$nK_n^{[p]} := \left[\int_{(0,1)} N'_{j+1,[p]} N'_{i+1,[p]}\right]_{i,j=1}^{n+p-2},$$

$$H_n^{[p]} := \left[\int_{(0,1)} N'_{j+1,[p]} N_{i+1,[p]}\right]_{i,j=1}^{n+p-2},$$

$$\frac{1}{n} M_n^{[p]} := \left[\int_{(0,1)} N_{j+1,[p]} N_{i+1,[p]}\right]_{i,j=1}^{n+p-2}.$$

From [25] we know that the above matrices have the following properties.

Lemma 2.1. *For every $p \ge 1$ and $n \ge 2$,*

- *$K_n^{[p]}$ is Symmetric Positive Definite (SPD) and $\|K_n^{[p]}\|_\infty \le 4p$;*
- *$H_n^{[p]}$ is skew-symmetric and $\|H_n^{[p]}\|_\infty \le 2$;*
- *$M_n^{[p]}$ is SPD, $\|M_n^{[p]}\|_\infty \le 1$ and $\exists\, C^{[p]} > 0$, depending only on p, such that $\lambda_{\min}(M_n^{[p]}) > C^{[p]}$.*

Although the argumentation in the 2D case follows more or less the same pattern as in the 1D case, we will briefly describe it, both for the sake of completeness and for illustrating the strict analogies between the 1D and 2D setting. Given any two functions $f, g : [a,b] \to \mathbb{R}$, we denote by $f \otimes g$ the tensor-product function

$$f \otimes g : [a,b]^2 \to \mathbb{R}, \quad (f \otimes g)(\theta_1, \theta_2) = f(\theta_1) g(\theta_2).$$

Without loss of clarity, we can use the same symbol $\otimes$ for the tensor (or Kronecker) product of matrices (see [7]):

$$X \otimes Y = \left[[x_{i_1,j_1} y_{i_2,j_2}]_{i_2,j_2=1}^{m_2} \right]_{i_1,j_1=1}^{m_1},$$

with $X = [x_{i,j}]_{i,j=1}^{m_1}$ and $Y = [y_{i,j}]_{i,j=1}^{m_2}$.

We now approximate the weak solution u of (2.1) by means of the approximation space $\mathcal{W}$ chosen as a space spanned by tensor-product B-splines. More precisely, we set $\mathcal{W} = \mathcal{W}_{n_1,n_2}^{[p_1,p_2]}$, for some $p_1, p_2 \geq 1,\ n_1, n_2 \geq 2$, where

$$\mathcal{W}_{n_1,n_2}^{[p_1,p_2]} := \langle N_{j_1,[p_1]} \otimes N_{j_2,[p_2]} :\ j_1 = 2, \dots, n_1 + p_1 - 1,\ j_2 = 2, \dots, n_2 + p_2 - 1 \rangle,$$

and the univariate functions $N_{j,[p]}$ are described in Definition 2.1, see also [25, Section 5]. The tensor-product B-splines form a basis for $\mathcal{W}_{n_1,n_2}^{[p_1,p_2]}$, and we order them in the same way as in [25, Eq. (85)], namely

$$\left[\left[N_{j_1,[p_1]} \otimes N_{j_2,[p_2]} \right]_{j_1=2,\dots,n_1+p_1-1} \right]_{j_2=2,\dots,n_2+p_2-1}.$$

Then, we obtain in (2.4) the following stiffness matrix, see [25, Section 5.1]:

$$A_{n_1,n_2}^{[p_1,p_2]} := K_{n_1,n_2}^{[p_1,p_2]} + \frac{\beta_1}{n_2} M_{n_2}^{[p_2]} \otimes H_{n_1}^{[p_1]} + \frac{\beta_2}{n_1} H_{n_2}^{[p_2]} \otimes M_{n_1}^{[p_1]} + \frac{\gamma}{n_1 n_2} M_{n_2}^{[p_2]} \otimes M_{n_1}^{[p_1]},$$

where

$$K_{n_1,n_2}^{[p_1,p_2]} := \frac{n_1}{n_2} M_{n_2}^{[p_2]} \otimes K_{n_1}^{[p_1]} + \frac{n_2}{n_1} K_{n_2}^{[p_2]} \otimes M_{n_1}^{[p_1]},$$

and the matrices $K_n^{[p]}$, $H_n^{[p]}$, $M_n^{[p]}$ have been previously defined for all $p \geq 1$ and $n \geq 2$ (see Lemma 2.1 for their main properties).

Remark 2.1. *By Lemma 2.1 and by the fact that $X \otimes Y$ is SPD whenever X, Y are SPD, we know that $A_{n_1,n_2}^{[p_1,p_2]}$ is SPD for all $p_1, p_2 \geq 1$ and $n_1, n_2 \geq 2$, provided that $\boldsymbol{\beta} = \mathbf{0}$.*

3. Spectral analysis, Toeplitz matrices, and the symbol

We start with the definition of spectral distribution, according to a symbol, of a given sequence of matrices, and then we define d-level Toeplitz matrices generated by a d-variate function g. We focus on d-level Toeplitz matrices, because our coefficient matrices in (2.4) are of d-level Toeplitz type up to a correction, whose rank is of order $m^{\frac{d-1}{d}}$ with m being the global matrix size. It turns out that g is the symbol of the Toeplitz matrices generated by g, and the matrices in (2.4) have the same symbol like their Toeplitz part (see [25] for details).

3.1. Spectral analysis and multilevel Toeplitz matrices

We denote by μ_d the Lebesgue measure in $\mathbb{R}^d$, and by $C_c(\mathbb{C}, \mathbb{C})$ the space of continuous functions $F : \mathbb{C} \to \mathbb{C}$ with compact support. We also set $\boldsymbol{\theta} := (\theta_1, \dots, \theta_d)$.

Definition 3.1. *Let $\{X_n\}$ be a sequence of matrices with increasing size ($X_n \in \mathbb{C}^{m_n \times m_n}$ with $m_n < m_{n+1}$, $\forall n$) and let $f : D \subset \mathbb{R}^d \to \mathbb{C}$ be a measurable function defined on the measurable set D with $0 < \mu_d(D) < \infty$. We say that $\{X_n\}_n$ is distributed like f in the sense of the eigenvalues, and we write $\{X_n\}_n \sim_\lambda f$, if $\forall F \in C_c(\mathbb{C}, \mathbb{C})$,*

$$\lim_{n \to \infty} \frac{1}{m_n} \sum_{j=1}^{m_n} F(\lambda_j(X_n)) = \frac{1}{\mu_d(D)} \int_D F(f(\boldsymbol{\theta}))\, \mathrm{d}\boldsymbol{\theta}. \tag{3.1}$$

The function f is referred to as the symbol of the sequence of matrices $\{X_n\}_n$.

Remark 3.1. *The informal meaning behind the above definition is the following. If f is continuous and $\left\{x_j^{(m_n)},\ j=1,\dots,m_n\right\}$ is an equispaced grid on D, then a suitable ordering of the pairs $\left\{\left(x_j^{(m_n)},\lambda_j(X_n)\right)\right\}$ reconstructs approximately the surface $(t,f(t))$. For instance, if f is continuous, $d=1$, $m_n=n$, and $D=[a,b]$, then the eigenvalues of X_n are approximately equal to $f(a+j(b-a)/n)$, $j=1,\dots,n$, for n large enough. Analogously, if f is continuous, $d=2$, $m_n=n^2$, and $D=[a_1,b_1]\times[a_2,b_2]$, then the eigenvalues of X_n are approximately equal to $f(a_1+j(b_1-a_1)/n, a_2+k(b_2-a_2)/n)$, $j,k=1,\dots,n$, for n large enough. Of course, this can be easily extended to the d-dimensional setting. Finally, we remark that the notion can be applied to a more general (Peano-Jordan measurable) domain, by taking the sampling points uniformly distributed in the domain.*

Now we consider the important case of d-level Toeplitz matrices.

Definition 3.2. *Given a d-variate function $g:[-\pi,\pi]^d\to\mathbb{R}$ in $L^1([-\pi,\pi]^d)$ and a multi-index $\boldsymbol{m}:=(m_1,\dots,m_d)\in\mathbb{N}^d$, $T_{\boldsymbol{m}}(g)$ is the d-level Toeplitz matrix of partial orders $m_1,\dots,m_d$ (and order $\prod_{j=1}^d m_j$) associated with g, i.e.,*

$$T_{\boldsymbol{m}}(g):=\left[\cdots\left[\left[g_{i_1-j_1,i_2-j_2,\dots,i_d-j_d}\right]_{i_d,j_d=1}^{m_d}\right]_{i_{d-1},j_{d-1}=1}^{m_{d-1}}\cdots\right]_{i_1,j_1=1}^{m_1},$$

where $g_{i_1,i_2,\dots,i_d}$, $i_1,i_2,\dots,i_d\in\mathbb{Z}$, are the Fourier coefficients of g,

$$g_{i_1,i_2,\dots,i_d}:=\frac{1}{(2\pi)^d}\int_{[-\pi,\pi]^d}g(\boldsymbol{\theta})\mathrm{e}^{-\mathrm{i}(i_1\theta_1+i_2\theta_2+\cdots+i_d\theta_d)}\mathrm{d}\boldsymbol{\theta}.$$

The function g is called the generating function of the Toeplitz family $\{T_{\boldsymbol{m}}(g)\}_{\boldsymbol{m}\in\mathbb{N}^d}$.

By the Szegö-Tilli theorem [38], a distribution relation holds for $\{T_{\boldsymbol{m}}(g)\}_{\boldsymbol{m}\in\mathbb{N}^d}$ and in fact, $\forall F\in C_c(\mathbb{C},\mathbb{C})$, we have

$$\lim_{\boldsymbol{m}\to\infty}\frac{1}{m_1\cdots m_d}\sum_{j=1}^{m_1\cdots m_d}F[\lambda_j(T_{\boldsymbol{m}}(g))]=\frac{1}{(2\pi)^d}\int_{[-\pi,\pi]^d}F[g(\boldsymbol{\theta})]\,\mathrm{d}\boldsymbol{\theta},\tag{3.2}$$

where for a multi-index $\boldsymbol{m}$, '$\boldsymbol{m}\to\infty$' means that '$\min(m_1,\dots,m_d)\to\infty$'. For this reason, g is called the symbol of the Toeplitz family $\{T_{\boldsymbol{m}}(g)\}_{\boldsymbol{m}\in\mathbb{N}^d}$ and, according to Definition 3.1, we write $\{T_{\boldsymbol{m}}(g)\}_{\boldsymbol{m}\in\mathbb{N}^d}\sim_\lambda g$.

Suppose that $g:[-\pi,\pi]^d\to\mathbb{R}$ is continuous over $[-\pi,\pi]^d$ and symmetric in each variable, in the sense that $g(\varepsilon_1\theta_1,\dots,\varepsilon_d\theta_d)=g(\theta_1,\dots,\theta_d)$ for $(\theta_1,\dots,\theta_d)\in[-\pi,\pi]^d$ and $(\varepsilon_1,\dots,\varepsilon_d)\in\{-1,1\}^d$. Then, the right-hand side of (3.2) coincides with

$$\frac{1}{\pi^d}\int_{[0,\pi]^d}F[g(\boldsymbol{\theta})]\,\mathrm{d}\boldsymbol{\theta},$$

so that the symbol g can be considered, at the same time, in two different domains, i.e. $[-\pi,\pi]^d$ and $[0,\pi]^d$. This example shows the non-uniqueness of the symbol. In fact, there exist infinitely many symbols for every fixed sequence, see e.g. [14, 35]. This fact is not a weak point of the theory, because we can have more degrees of freedom for describing important global spectral properties, as sketched in the next subsection.

3.2. How to use the symbol? A basic guide to the user

We now explain some basic information that we can extract from a symbol, and we sketch its use from a practical viewpoint. We mainly focus our attention on a perturbed Toeplitz setting, because our IgA matrices can be regarded as small rank perturbations of Toeplitz matrices with a real-valued symbol of (trigonometric) polynomial type, see [25] and Section 4.

3.2.1. Counting the eigenvalues belonging to a given interval

The starting point of our reasoning is Definition 3.1 and especially the subsequent Remark 3.1. Let $a < b$, let $\{X_n\}$ be a sequence of Hermitian matrices of size m_n distributed like f, and let $E_n([a,b])$ be the number of eigenvalues of X_n belonging to the interval $[a,b]$. Then, relation (3.1) implies

$$E_n([a,b]) = I[a,b]m_n + o(m_n), \tag{3.3}$$

with

$$I[a,b] = \mu_d(\{\boldsymbol{\theta} \in D : \ f(\boldsymbol{\theta}) \in [a,b]\})/\mu_d(D),$$

if

$$0 = \mu_d\{\boldsymbol{\theta} \in D : \ f(\boldsymbol{\theta}) = a\} = \mu_d\{\boldsymbol{\theta} \in D : \ f(\boldsymbol{\theta}) = b\}. \tag{3.4}$$

Regarding the hypothesis in equation (3.4), we observe that it is never violated when f is a non-constant trigonometric polynomial. However, it can be violated, for a general measurable function f, only for countably many values of a or b.

The expression of the error term $o(m_n)$ can be better estimated under specific circumstances. For example, if $X_n = T_{m_n}(f)$, $d = 1$, and f is a real-valued trigonometric polynomial, then the error term $o(m_n)$ can be replaced by a constant linearly depending on the degree of f (this can be deduced by using Cauchy interlacing arguments, see [31]). The same arguments hold for our IgA matrices with $d = 1$, because they are a constant rank correction of given Toeplitz matrices $T_{m_n}(f_p)$, where f_p is a trigonometric polynomial of degree p and where the rank of the correction matrix is proportional to p (see [25]).

Formula (3.3) is of interest e.g. when $a = 0$, $b = \epsilon \ll 1$ for having a good guess of the size of the eigenspace related to small eigenvalues $\lambda \leq \epsilon \ll 1$. In fact, this subspace is responsible for the ill-conditioning of the matrix and for the slow convergence of general purpose iterative solvers (see [3]).

3.2.2. Extremal eigenvalues and conditioning in a perturbed Toeplitz setting

We can use again the analytic properties of the symbol also for understanding the behavior of the extremal eigenvalues and of the conditioning, at least in a Toeplitz setting (see e.g. [9, 24, 31, 32] and references therein). Let $\boldsymbol{m_n} := (m_{n_1}, \ldots, m_{n_d})$, then we take $X_{\boldsymbol{n}} = T_{\boldsymbol{m_n}}(f)$, assuming f to be real-valued, nonnegative, bounded, and with a finite number of zeros of order $\alpha_1, \ldots, \alpha_r$. Let $\alpha := \max_{1 \leq i \leq r} \alpha_i$, and let $\kappa(\cdot)$ denote the condition number of its argument, and $m_{\boldsymbol{n}} := m_{n_1} \cdots m_{n_d}$, then

- $\lambda_{\min}(T_{\boldsymbol{m_n}}(f)) \sim [m_{\boldsymbol{n}}]^{-\alpha/d}$;
- $\lim_{\boldsymbol{n}\to\infty} \lambda_{\max}(T_{\boldsymbol{m_n}}(f)) = \operatorname{ess\,sup} f$;
- $\kappa(T_{\boldsymbol{m_n}}(f)) \sim [m_{\boldsymbol{n}}]^{\alpha/d}$.

These properties are shared by our matrices $A_n^{[p]}$ and $A_{n_1,n_2}^{[p_1,p_2]}$, where the role of the symbol f is played by f_p and $f_{p_1,p_2}^{(\nu_1,\nu_2)}$, respectively. Therefore, by looking at the expression of the symbols, the relevant quantities are $r = 1$, $\alpha = \alpha_1 = 2$ (see [25] and Section 4).

3.2.3. Eigenvectors vs frequencies in a perturbed Toeplitz setting

This subsection is the most interesting from the viewpoint of designing optimal and robust iterative solvers. For the sake of simplicity and also for the purposes of our paper, we can restrict the attention to the case of sequences of matrices belonging to the algebra generated by Toeplitz sequences, modulo zero distributed sequences. We denote by $\mathcal{T}$ this

set of sequences. In this context, a lot can be said concerning the approximate structure of the eigenspaces in terms of frequencies.

Roughly speaking, let $d = 1$, and let f be a smooth, real-valued, even function, such that it is the symbol of a sequence $\{X_n\}_n \in \mathcal{T}$ of real symmetric matrices of size m_n, then the eigenvalues $\lambda_j(X_n)$, $j = 1, \ldots, m_n$, behave as the uniform sampling

$$f\left(\pi j/(m_n+1)\right), \quad j = 1, \ldots, m_n,$$

and the related eigenvectors behave as the following set of frequency vectors (vectors associated to a famous sine transform)

$$\mathbf{v}_j^{(m_n)} = \left(\sin\left(jx_k\right)\right)_{k=1}^{m_n}, \qquad x_k = \pi k/(m_n+1).$$

The statement above is quite vague but it can be made more precise, without using technicalities (see [10, 42] for a rigorous analysis). If we are interested in the eigenvectors associated to eigenvalues in the interval $[a, b]$, then we know that this subspace has dimension $I[a, b]m_n + o(m_n)$ and it is approximately described by

$$\operatorname{span}\left\{\mathbf{v}_j^{(m_n)} : \ \pi j/(m_n+1) \in \{\theta \in [0,\pi] : \ f(\theta) \in [a,b]\}\right\}. \tag{3.5}$$

From the relation above it can be seen that a zero of the symbol at $\theta = 0$ implies that the ill-conditioned subspace is related to low frequencies, while a zero of the symbol at π implies that the ill-conditioned subspace is related to high frequencies.

If $d > 1$ then we use tensor-like arguments and the same conclusions hold. For instance, if $d = 2$ and we are interested in the eigenvectors associated to eigenvalues in the interval $[a, b]$, then we know that this subspace has dimension $I[a, b]m_{\boldsymbol{n}} + o(m_{\boldsymbol{n}})$ ($m_{\boldsymbol{n}} = m_{n_1}m_{n_2}$, m_{n_1}, m_{n_2} partial dimensions) and it is approximately described by

$$\operatorname{span}\left\{\mathbf{v}_{j_1}^{(m_{n_1})} \otimes \mathbf{v}_{j_2}^{(m_{n_2})} : \ (\pi j_1/(m_{n_1}+1), \pi j_2/(m_{n_2}+1)) \in f^{-1}([a,b])\right\}, \tag{3.6}$$

with $f^{-1}([a, b]) := \{(\theta_1, \theta_2) \in [0, \pi]^2 : \ f(\theta_1, \theta_2) \in [a, b]\}$.

4. The symbol of IgA Galerkin matrices

In this section we describe the symbol (and the main analytic features) related to the Galerkin matrices based on B-splines. We first summarize from [25, Section 4] the symbol and some of its properties in the 1D case. These results are the building blocks for the multivariate setting, thanks to the tensor-product structure. Then, we discuss the symbol in the bivariate case (see [25, Section 5]), and finally we briefly sketch the d-dimensional version.

4.1. The symbol of the sequence $\{\frac{1}{n}A_n^{[p]}\}_n$

For $p \geq 0$, let $\phi_{[p]}$ be the cardinal B-spline of degree p over the uniform knot sequence $\{0, 1, \ldots, p+1\}$, which is defined recursively as follows [8]:

$$\phi_{[0]}(t) := \begin{cases} 1, & \text{if } t \in [0,1), \\ 0, & \text{elsewhere,} \end{cases}$$

and

$$\phi_{[p]}(t) := \frac{t}{p}\phi_{[p-1]}(t) + \frac{p+1-t}{p}\phi_{[p-1]}(t-1), \quad p \geq 1.$$

We point out that the 'central' basis functions $N_{i,[p]}(x)$, $i = p+1, \ldots, n$ in Definition 2.1 are cardinal B-splines, namely

$$N_{i,[p]}(x) = \phi_{[p]}(nx - i + p + 1), \quad i = p+1, \ldots, n.$$

Let us denote by $\ddot{\phi}_{[p]}(t)$ the second derivative of $\phi_{[p]}(t)$ with respect to its argument t (for $p \geq 3$). For $p \geq 0$, let $h_p : [-\pi, \pi] \to \mathbb{R}$,

$$h_0(\theta) := 1, \qquad h_p(\theta) := \phi_{[2p+1]}(p+1) + 2\sum_{k=1}^{p} \phi_{[2p+1]}(p+1-k)\cos(k\theta), \tag{4.1}$$

and, for $p \geq 1$, let $f_p : [-\pi, \pi] \to \mathbb{R}$,

$$f_p(\theta) := -\ddot{\phi}_{[2p+1]}(p+1) - 2\sum_{k=1}^{p} \ddot{\phi}_{[2p+1]}(p+1-k)\cos(k\theta). \tag{4.2}$$

Using the fact that the sum of all singular values of $\frac{1}{n}A_n^{[p]} - T_{n+p-2}(f_p)$ is bounded from above by a constant independent of n, it has been proved in [25, Theorem 12] that, for each fixed $p \geq 1$,

$$\lim_{n\to\infty} \frac{1}{n+p-2} \sum_{j=1}^{n+p-2} F\left(\lambda_j\left(\frac{1}{n}A_n^{[p]}\right)\right) = \frac{1}{2\pi}\int_{-\pi}^{\pi} F(f_p(\theta))\mathrm{d}\theta, \quad \forall F \in C_c(\mathbb{C}, \mathbb{C}).$$

Hence, f_p is the symbol of the sequence of scaled matrices $\{\frac{1}{n}A_n^{[p]}\}_n$, and

$$\{\frac{1}{n}A_n^{[p]}\}_n \sim_\lambda f_p,$$

according to Definition 3.1. We note that f_p is symmetric on $[-\pi, \pi]$, so f_p restricted to $[0, \pi]$ is also a symbol for $\{\frac{1}{n}A_n^{[p]}\}_n$. The symbol f_p is independent of β and γ, and possesses the properties collected in Lemma 4.1, see [25, Section 3]. Recall that the modulus of the Fourier transform of the cardinal B-spline $\phi_{[p]}$ is given by

$$\left|\widehat{\phi_{[p]}}(\theta)\right|^2 = \left(\frac{2-2\cos\theta}{\theta^2}\right)^{p+1}.$$

Lemma 4.1. *The following properties hold for all $p \geq 1$ and $\theta \in [-\pi, \pi]$:*

1. $f_p(\theta) = (2 - 2\cos\theta)h_{p-1}(\theta)$;
2. $h_{p-1}(\theta) = \sum_{k\in\mathbb{Z}} \left|\widehat{\phi_{[p-1]}}(\theta + 2k\pi)\right|^2$;
3. $\left(\frac{4}{\pi^2}\right)^p \leq h_{p-1}(\theta) \leq h_{p-1}(0) = 1$.

Note that the properties in Lemma 4.1 have been proved in [25, Lemma 7 and Remark 2] for $p \geq 2$, but it can be checked that they also hold for $p = 1$.

Figure 1 shows the graph of f_p normalized by its maximum M_{f_p}, for $p = 1, \ldots, 5$. The value $f_p(\pi)/M_{f_p}$ decreases exponentially to zero as $p \to \infty$, see Table 1. This is formally proved in [16]. From a numerical viewpoint, we can say that, for large p, the normalized symbol f_p/M_{f_p} possesses two zeros over $[0, \pi]$: one in $\theta = 0$ and the other in the corresponding

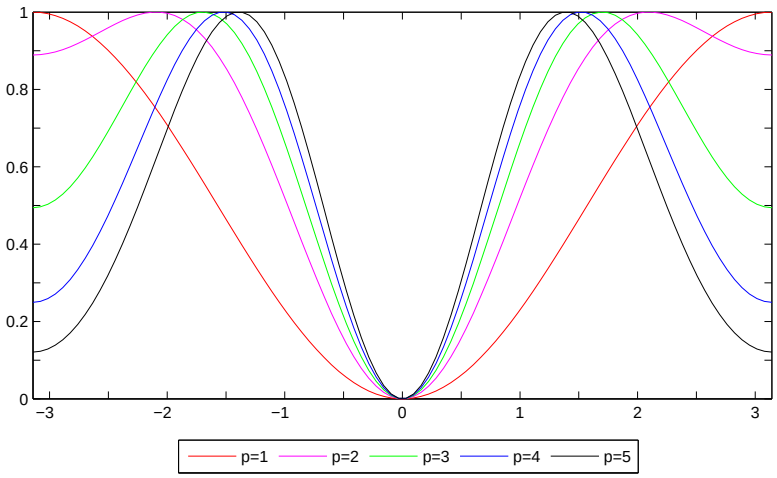

Figure 1: Graph of f_p/M_{f_p} for $p = 1, \ldots, 5$.

p	1	2	3	4	5	6	7	8	9	10
$f_p(\pi)/M_{f_p}$	1.000	0.889	0.494	0.249	0.129	0.057	0.026	0.012	0.005	0.002

Table 1: Values of $f_p(\pi)/M_{f_p}$ for $p = 1, \ldots, 10$.

mirror point $\theta = \pi$. Because of this, we expect intrinsic difficulties, in particular a slow (though optimal) convergence rate, when solving, for large p, a linear system of the form $\frac{1}{n}A_n^{[p]}\mathbf{u} = \mathbf{b}$ by means of the two-grid method described in Section 5.2 with as projector (5.8), which halves the size of the original system at each iteration. Possible ways to overcome this problem are choosing a different size reduction at the lower level and/or adopting a multi-iterative strategy involving a variation of the smoothers. Both these possibilities are discussed in Section 5 and tested numerically in Section 6.2.

A (numerical) pathological behavior and its explanation

Finally, as an example, we consider the problem of computing the 100 smallest eigenvalues $0 < \lambda_1^{(n,p)} \leq \cdots \leq \lambda_{100}^{(n,p)}$ and the related eigenvectors $\mathbf{v}_1^{(n,p)}, \ldots, \mathbf{v}_{100}^{(n,p)}$ of $A_n^{[p]}$.[1] From the theory of elliptic operators, we know that if the finesse parameter is small enough (say $n \geq 10^4$), then we expect that $\mathbf{v}_{100}^{(n,p)}$ is a good approximation of the smooth eigenfunction $\mathbf{v}_{100}(x)$.

Unfortunately, if p is large enough, then $f_p(\pi) < \lambda_{100}^{(n,p)}$ because $f_p(\pi)$ converges monotonically to zero (and with exponential speed). Consequently, the horizontal line defined as $y = \lambda_{100}^{(n,p)}$ intersects the graph of $f_p(\theta)$ at two points θ_{low} and θ_{high}, where $0 < \theta_{\text{low}} \ll \pi/2$ and θ_{high} close to π.

According to the discussion in Section 3.2.3 on Toeplitz symbols and eigenspaces (see equations (3.5) and (3.6)), this means that the computed eigenvector $\mathbf{v}_{100}^{(n,p)}$ could be a linear

[1] For simplicity and clarity we only consider this example in the 1D case, but the same applies for every dimensionality.

combination of two eigenvectors: one being low frequency (the correct one approximating the eigenfunction $\mathbf{v}_{100}(x)$) and one being very high frequency which represents a spurious eigenvector introduced by our approximation and in fact depending on p. In reality, this second vector is due to the non-canonical behavior of $f_p(\theta)$ in a neighborhood of π, for large p.

As simply indicated by the latter example, the symbol can give clean explanations for behaviors or pathological phenomena which arise when numerical methods are applied to our approximating matrices.

4.2. The symbol of the sequence $\{A^{[p_1,p_2]}_{\nu_1 n,\nu_2 n}\}_n$

For $p_1, p_2 \geq 1$ and $\nu_1, \nu_2 \in \mathbb{Q}_+ := \{r \in \mathbb{Q} : \ r > 0\}$, we define the function

$$f^{(\nu_1,\nu_2)}_{p_1,p_2} : [-\pi,\pi]^2 \to \mathbb{R}, \quad f^{(\nu_1,\nu_2)}_{p_1,p_2} := \frac{\nu_1}{\nu_2} h_{p_2} \otimes f_{p_1} + \frac{\nu_2}{\nu_1} f_{p_2} \otimes h_{p_1}, \tag{4.3}$$

see (4.1)–(4.2) for the definition of h_p and f_p. From now on we always assume that $n \in \mathbb{N}$ is chosen such that $n\boldsymbol{\nu} \in \mathbb{N}^2$, where $\boldsymbol{\nu} := (\nu_1, \nu_2)$. Consider the sequence of matrices

$$A^{[p_1,p_2]}_{\nu_1 n,\nu_2 n} = K^{[p_1,p_2]}_{\nu_1 n,\nu_2 n} + \frac{\beta_1}{\nu_2 n} M^{[p_2]}_{\nu_2 n} \otimes H^{[p_1]}_{\nu_1 n} + \frac{\beta_2}{\nu_1 n} H^{[p_2]}_{\nu_2 n} \otimes M^{[p_1]}_{\nu_1 n} + \frac{\gamma}{\nu_1 \nu_2 n^2} M^{[p_2]}_{\nu_2 n} \otimes M^{[p_1]}_{\nu_1 n},$$

with n varying in the set of indices where $n \geq 2$ and $\nu_1 n, \nu_2 n \geq 2$. It was proved in [25, Section 5.2] that, $\forall F \in C_c(\mathbb{C}, \mathbb{C})$,

$$\lim_{n\to\infty} \frac{1}{N} \sum_{j=1}^{N} F\left(\lambda_j\left(A^{[p_1,p_2]}_{\nu_1 n,\nu_2 n}\right)\right) = \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} F(f^{(\nu_1,\nu_2)}_{p_1,p_2}(\theta_1,\theta_2)) \mathrm{d}\theta_1 \mathrm{d}\theta_2,$$

with $N := (\nu_1 n + p_1 - 2)(\nu_2 n + p_2 - 2)$, and so $f^{(\nu_1,\nu_2)}_{p_1,p_2}$ is the symbol of the sequence $\{A^{[p_1,p_2]}_{\nu_1 n,\nu_2 n}\}_n$. The symbol $f^{(\nu_1,\nu_2)}_{p_1,p_2}$ is independent of $\boldsymbol{\beta}$ and γ, and possesses the following properties (consequences of Lemma 4.1).

Lemma 4.2. *Let $p_1, p_2 \geq 1$ and $\nu_1, \nu_2 \in \mathbb{Q}_+$. Then, $\forall (\theta_1, \theta_2) \in [-\pi,\pi]^2$,*

$$f^{(\nu_1,\nu_2)}_{p_1,p_2}(\theta_1,\theta_2) \geq \left(\frac{4}{\pi^2}\right)^{p_1+p_2+1} \min\left(\frac{\nu_1}{\nu_2}, \frac{\nu_2}{\nu_1}\right) (4 - 2\cos\theta_1 - 2\cos\theta_2),$$

$$f^{(\nu_1,\nu_2)}_{p_1,p_2}(\theta_1,\theta_2) \leq \max\left(\frac{\nu_1}{\nu_2}, \frac{\nu_2}{\nu_1}\right) (4 - 2\cos\theta_1 - 2\cos\theta_2).$$

Let $M_{f^{(\nu_1,\nu_2)}_{p_1,p_2}} := \max_{\boldsymbol{\theta}\in[0,\pi]^2} f^{(\nu_1,\nu_2)}_{p_1,p_2}(\boldsymbol{\theta})$. By Lemma 4.2, the normalized symbol $f^{(\nu_1,\nu_2)}_{p_1,p_2}/M_{f^{(\nu_1,\nu_2)}_{p_1,p_2}}$ has only one actual zero at $\boldsymbol{\theta} = \mathbf{0}$. However, when p_1, p_2 are large, it also has infinitely many 'numerical zeros' over $[0,\pi]^2$, located at the 'π-edge points'

$$\{(\theta_1, \pi) : \ 0 \leq \theta_1 \leq \pi\} \cup \{(\pi, \theta_2) : \ 0 \leq \theta_2 \leq \pi\}, \tag{4.4}$$

because, as proved in [16], we have

$$f^{(\nu_1,\nu_2)}_{p_1,p_2}(\theta_1,\pi) \leq \frac{1}{2^{p_1-2}} M_{f^{(\nu_1,\nu_2)}_{p_1,p_2}}, \qquad f^{(\nu_1,\nu_2)}_{p_1,p_2}(\pi,\theta_2) \leq \frac{1}{2^{p_2-2}} M_{f^{(\nu_1,\nu_2)}_{p_1,p_2}}.$$

Because of this unpleasant property, the two-grid schemes that we are going to devise for the matrix $A^{[p_1,p_2]}_{\nu_1 n,\nu_2 n}$ are expected to show a bad (though optimal) convergence rate when either p_1 or p_2 is large. In analogy with the one-dimensional setting and taking into account Section 5.2, we can bypass the problem by adopting a multi-iterative strategy involving the PCG as smoother. The numerical tests are reported in Section 7.2.

4.3. The symbol in the d-dimensional case

Here we just give a sketch of the d-dimensional case, $d \geq 1$. For $p_1, \ldots, p_d \geq 1$ and $\nu_1, \ldots, \nu_d \in \mathbb{Q}_+$, we define the function

$$f_{p_1,\ldots,p_d}^{(\nu_1,\ldots,\nu_d)} : [-\pi,\pi]^d \to \mathbb{R}, \quad f_{p_1,\ldots,p_d}^{(\nu_1,\ldots,\nu_d)} := \sum_{j=1}^{d} c_j(\boldsymbol{\nu}) f_{p_j}(\theta_j) \prod_{k=1,k\neq j}^{d} h_{p_k}(\theta_k), \tag{4.5}$$

where $c_j(\boldsymbol{\nu})$ are positive constants depending only on $\boldsymbol{\nu} := (\nu_1, \ldots, \nu_d)$, and where the expressions of the basic symbols h_p and f_p are given in (4.1)–(4.2). Under the assumption that $n\boldsymbol{\nu} \in \mathbb{N}^d$, the sequence of scaled matrices $\{n^{d-2} A_{\nu_1 n,\ldots,\nu_d n}^{[p_1,\ldots,p_d]}\}$ approximating the general PDE in (2.1) is distributed in the eigenvalue sense like $f_{p_1,\ldots,p_d}^{(\nu_1,\ldots,\nu_d)}$.

The symbol $f_{p_1,\ldots,p_d}^{(\nu_1,\ldots,\nu_d)}$ possesses the following properties:

- it is independent of $\boldsymbol{\beta}$ and γ;
- it has a unique zero at $\boldsymbol{\theta} = \mathbf{0}$;
- it converges super-exponentially at zero in the variables $p_1, \ldots, p_d$, when they tend all to infinity, and so a large set of numerical zeros over $[0,\pi]^d$ occurs, for $p_1, \ldots, p_d$ large enough, located at the π-edges of the domain

$$\{\boldsymbol{\theta} : \exists j \in \{1,\ldots,d\} \ \text{ with } \ \theta_j = \pi\}. \tag{4.6}$$

Note that the set in (4.6) with $d = 2$ is exactly the set in (4.4), while for $d = 1$, as expected, it reduces to the point $\theta = \pi$.

5. Iterative solvers and the multi-iterative approach

In this section we review some basic iterative methods that we will consider further on:

1. classical stationary iterations (Richardson, Gauss-Seidel, the weighted versions, etc. [43]);
2. the PCG (Preconditioned Conjugate Gradient) method [3];
3. two-grid, V-cycle, W-cycle methods [41];
4. multi-iterative techniques [30].

We will present them in view of the multi-iterative approach [30], i.e., a way of combining different (basic) iterative solvers having complementary spectral behavior. We first recall some classical stationary iterative methods and explain the main idea of the multi-iterative approach. Then, we focus on two-grid and multigrid methods, and finally we end with a discussion on the PCG method in our IgA context.

5.1. Unity makes strength: the multi-iterative approach

Stationary iterative methods for solving a linear system $A\mathbf{u} = \mathbf{b}$ (with $A \in \mathbb{R}^{m\times m}$) can be written in the general form

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + M^{-1}(\mathbf{b} - A\mathbf{u}^{(k)}), \qquad k = 0, 1, \ldots, \tag{5.1}$$

where the matrix M^{-1} is taken to be an approximation to A^{-1} such that the product of M^{-1} and an arbitrary vector is easy to compute. By defining the iteration matrix $S := I - M^{-1}A$, we can reformulate the stationary iteration (5.1) as

$$\mathbf{u}^{(k+1)} = \mathcal{S}(\mathbf{u}^{(k)}) := S\mathbf{u}^{(k)} + (I - S)A^{-1}\mathbf{b}. \tag{5.2}$$

The error $\mathbf{e}^{(k+1)} := A^{-1}\mathbf{b} - \mathbf{u}^{(k+1)}$ is then given by $\mathbf{e}^{(k+1)} = S\mathbf{e}^{(k)}$, and its norm is quickly reduced if $\|S\|$ is much smaller than one.

Later on, in Sections 6.1 and 7.1, we will apply two particular examples of (5.2) as smoother in the multigrid environment: the relaxed Richardson iteration matrix $\overline{S}$ and the relaxed Gauss-Seidel iteration matrix $\widehat{S}$, i.e.,

$$\overline{S} := I - \omega A, \tag{5.3}$$

$$\widehat{S} := I - \left(\frac{1}{\omega}D - L\right)^{-1} A. \tag{5.4}$$

In both cases $\omega \in \mathbb{R}$, while D and L are the matrices coming from the splitting of A associated with the Gauss-Seidel method: D is the diagonal part of A and L is the lower triangular part of A, excluding the diagonal elements.

Let us now consider l different approximations to A^{-1}, say M_i^{-1}, $i = 1, \ldots, l$, and then l iterative methods with iteration matrices $S_i := I - M_i^{-1}A$, $i = 1, \ldots, l$. The following multi-iterative scheme can then be defined [30]:

$$\begin{aligned} \mathbf{u}^{(k,1)} &= S_1\mathbf{u}^{(k)} + \mathbf{b}_1, \\ \mathbf{u}^{(k,2)} &= S_2\mathbf{u}^{(k,1)} + \mathbf{b}_2, \\ &\vdots \\ \mathbf{u}^{(k+1)} &= S_l\mathbf{u}^{(k,l-1)} + \mathbf{b}_l, \end{aligned} \tag{5.5}$$

where $\mathbf{b}_i := M_i^{-1}\mathbf{b}$. Hence,

$$\mathbf{u}^{(k+1)} = S_l S_{l-1} \cdots S_2 S_1 \mathbf{u}^{(k)} + \mathbf{c},$$

and

$$\mathbf{c} = \mathbf{b}_l + S_l(\mathbf{b}_{l-1} + \cdots + S_3(\mathbf{b}_2 + S_2\mathbf{b}_1)\cdots).$$

Consequently, the errors $\mathbf{e}^{(k)} := A^{-1}\mathbf{b} - \mathbf{u}^{(k)}$ and $\mathbf{e}^{(k,i)} := A^{-1}\mathbf{b} - \mathbf{u}^{(k,i)}$, $k \geq 0$, $i = 1, \ldots, l-1$, are such that

$$\begin{aligned} \mathbf{e}^{(k,i)} &= S_i \cdots S_2 S_1 \mathbf{e}^{(k)}, \\ \mathbf{e}^{(k+1)} &= S_l \cdots S_2 S_1 \mathbf{e}^{(k)}. \end{aligned}$$

If S_i is highly contractive in a subspace $\mathcal{H}_i$ and if $S_{i-1}(\mathcal{L}_{i-1}) \subset \mathcal{H}_i$, where $\mathcal{L}_{i-1}$ is another subspace where S_{i-1} reduces slowly the norm of the error, then $\|S_i S_{i-1}\|$ can be much smaller than $\|S_i\|\|S_{i-1}\|$. This implies that multi-iterative methods can be fast, even when the basic iteration matrices have norms close to one, or even when the basic iterations are non-convergent.

5.2. Multigrid methods in a multi-iterative perspective

We consider again the linear system $A\mathbf{u} = \mathbf{b}$, which is of size m. We assume to have stationary iterative methods (the smoothers) as in (5.2) for the solution of the linear system, and a full-rank matrix (the projector) $P \in \mathbb{R}^{l \times m}$ with $l \leq m$. Then, the corresponding two-grid method for solving the linear system is given by the following algorithm.

Algorithm 5.1. *Given an approximation $\mathbf{u}^{(k)}$ to the solution $\mathbf{u} = A^{-1}\mathbf{b}$, the new approximation $\mathbf{u}^{(k+1)}$ is obtained by applying ν_{pre} steps of pre-smoothing as in (5.2) with iteration matrix S_{pre}, a coarse-grid correction, and ν_{post} steps of post-smoothing as in (5.2) with iteration matrix S_{post} as follows:*

1. $\nu_{\rm pre}$ *steps of pre-smoothing:* $\mathbf{u}^{(k,1)} = \mathcal{S}_{\rm pre}^{\nu_{\rm pre}}(\mathbf{u}^{(k)})$;
2. *compute the residual:* $\mathbf{r} = \mathbf{b} - A\mathbf{u}^{(k,1)}$;
3. *project the residual:* $\mathbf{r}^{(c)} = P\mathbf{r}$;
4. *compute the correction:* $\mathbf{e}^{(c)} = \left(P\,A\,P^T\right)^{-1}\mathbf{r}^{(c)}$;
5. *extend the correction:* $\mathbf{e} = P^T\mathbf{e}^{(c)}$;
6. *correct the initial approximation:* $\mathbf{u}^{(k,2)} = \mathbf{u}^{(k,1)} + \mathbf{e}$;
7. $\nu_{\rm post}$ *steps of post-smoothing:* $\mathbf{u}^{(k+1)} = \mathcal{S}_{\rm post}^{\nu_{\rm post}}(\mathbf{u}^{(k,2)})$.

Multigrid methods are obtained by applying recursively the two-grid scheme. In particular, we focus on the V-cycle and W-cycle methods.

Algorithm 5.2. *Let* $\mathbf{u}^{(k)}$ *be a given approximation to the solution* $\mathbf{u} = A^{-1}\mathbf{b}$. *In a V-cycle multigrid algorithm, the new approximation* $\mathbf{u}^{(k+1)}$ *is obtained by applying Algorithm 5.1 where step 4 is approximated by a recursive call to this algorithm, until the size of the matrix is* $O(1)$ *so that we can use a direct solver. In a W-cycle multigrid algorithm, step 4 in Algorithm 5.1 is replaced by two consecutive recursive calls.*

Steps 2–6 in Algorithm 5.1 define the so-called coarse-grid correction, which is a standard non-convergent iterative method with iteration matrix

$$CGC := I - P^T\left(P\,A\,P^T\right)^{-1}P\,A. \tag{5.6}$$

The iteration matrix of the two-grid scheme is denoted by $TG(S_{\rm pre}^{\nu_{\rm pre}}, S_{\rm post}^{\nu_{\rm post}}, P)$ and its explicit form is given by

$$S_{\rm post}^{\nu_{\rm post}} \cdot CGC \cdot S_{\rm pre}^{\nu_{\rm pre}}.$$

The iteration matrices of the V-cycle and W-cycle are defined in the same way by replacing $\left(P\,A\,P^T\right)^{-1}$ in (5.6) by an expression defined recursively (see [2]). Furthermore, when the pre-smoothing is not present, the two-grid iteration matrix is denoted by $TG(S_{\rm post}^{\nu_{\rm post}}, P)$.

We point out that two-grid (and multigrid) methods can be written in the general multi-iterative form (5.5) where $l = 2$ or $l = 3$. In this case, S_1 is the pre-smoothing operator, S_2 is the coarse-grid operator, and S_3 is the post-smoothing operator.

Interestingly enough, we observe that $\|S_2\| \geq 1$ because its spectral radius is equal to 1, while S_1 and S_3 are usually weakly contractive. However, as we will see later in Section 6.1, there are examples in which the best contraction factor of the whole multi-iterative (two-grid) scheme is achieved by choosing a non-convergent smoother. Therefore, it may happen that a very fast multi-iterative method is obtained by combining basic iterations that are all slowly convergent or even non-convergent.

5.3. Multi-iterative solvers vs spectral distributions

The main idea of the multi-iterative approach is to choose the different iteration matrices S_i, $i = 1, \ldots, l$, in the scheme (5.5) such that they have a complementary spectral behavior. Let us assume that S_i is highly contractive in a subspace $\mathcal{H}_i$, and weakly (or not) contractive in the complementary subspace $\mathcal{L}_i$. Then, the recipe for designing fast multi-iterative solvers is to choose the iteration matrices S_i such that

$$\oplus_{i=1}^{l}\mathcal{H}_i = \mathbb{C}^m.$$

This recipe is aesthetically beautiful and appealing, but it is totally unpractical if we are unable to identify l pairs of subspaces $(\mathcal{H}_i, \mathcal{L}_i)$, $i = 1, \ldots, l$, with the properties described above and such that $\mathcal{H}_i \oplus \mathcal{L}_i = \mathbb{C}^m$.

However, the matrices appearing in the IgA approximation of our model problem (2.1) can be considered as perturbations of Toeplitz structures (see [25]), and for them Sections 3.2.1 and 3.2.3 can guide us to identify such subspaces in terms of frequencies and to estimate their dimensions, see in particular relationships (3.3)–(3.6).

Let us now illustrate this concept in the case of the d-dimensional discrete Laplacian on $[0,1]^d$ obtained by standard Finite Differences. Note that the discretization matrices are the same in the Galerkin approach based on multilinear tensor-product B-splines ($p_1 = \cdots = p_d = 1$). It is easy to see that such a matrix has a pure Toeplitz structure with its generating function given by

$$f(\boldsymbol{\theta}) = f_{\mathrm{FD}}(\boldsymbol{\theta}) = \sum_{j=1}^{d} (2 - 2\cos(\theta_j)), \quad \boldsymbol{\theta} \in [0,\pi]^d. \tag{5.7}$$

Now consider a multigrid method in the framework of a multi-iterative solver. It is composed by three iterations ($l = 3$): a pre-smoothing given by the Richardson method (5.3) with parameter ω_{pre} (iteration matrix S_1), a coarse-grid iteration with iteration matrix S_2 defined in (5.6), and a post-smoothing given by the Richardson method with parameter ω_{post} (iteration matrix S_3). The associated coarse-grid iteration described in [20, 21] with the projector in (5.8) is designed in such a way that the related iteration is not convergent globally, but strongly reduces the error in low frequencies. Now, $S_1 = I - \omega_{\mathrm{pre}} T_m(f) = T_m(1 - \omega_{\mathrm{pre}} f)$ and $S_3 = I - \omega_{\mathrm{post}} T_m(f) = T_m(1 - \omega_{\mathrm{post}} f)$. If we choose $\omega_{\mathrm{pre}} = \|f\|_\infty^{-1}$ it is easily seen that the symbol of the iteration matrix is equal to $1 - f/\|f\|_\infty$ which is maximal at $\boldsymbol{\theta} = (0,\ldots,0)$ and attains its minimum at $\boldsymbol{\theta} = (\pi,\ldots,\pi)$. As a consequence, the pre-smoothing iteration is fast convergent in the high frequencies and it is slow in the low frequencies.

In fact, if we consider a two-grid (and the related V-cycle multigrid) with the latter coarse-grid correction operator and the latter pre-smoother, then we already obtain an optimal method (see [34, 37]), even though the two basic iterations are very slow or non-convergent.

However, at this point, we understand the machinery, and hence, if we desire to accelerate further the global multi-iterative method, then we can consider a post-smoothing iteration which may be slowly convergent both in the very low and very high frequencies but it is very fast in a space of 'intermediate' frequencies. The choice is obtained by setting $\omega_{\mathrm{post}} = 2\|f\|_\infty^{-1}$ so that $S_3 = T_m(1 - 2f/\|f\|_\infty)$. It is interesting to remark that the symbol $|1 - 2f(\boldsymbol{\theta})/\|f\|_\infty|$ evaluated at $\boldsymbol{\theta} = (0,\ldots,0)$ and $\boldsymbol{\theta} = (\pi,\ldots,\pi)$ is equal to 1. Therefore, the method is slowly convergent (moduli of the eigenvalues close to 1) both in high and slow frequencies, but the symbol is very small in absolute value in regions of $[0,\pi]^d$ associated to intermediate frequencies.

This multi-iterative method is indeed extremely fast, as shown in [37]. We will use these guiding ideas in our choice of the solvers for the IgA matrices.

5.4. Choice of the projector in our two-grid and multigrid methods

We now look for an appropriate projector in the coarse-grid correction (5.6) in order to address our specific linear systems. Since the matrices of interest can be considered as perturbations of d-level Toeplitz matrices (see [25]), we follow the approach in [18, 34]. Let $\boldsymbol{m} := (m_1,\ldots,m_d) \in \mathbb{N}^d$ be a multi-index satisfying certain additional constraints to be seen later. We consider a projector of the form

$$Z_{\boldsymbol{m}} \cdot T_{\boldsymbol{m}}(z_d),$$

where $Z_{\boldsymbol{m}} := Z_{m_1} \otimes \cdots \otimes Z_{m_d}$ and $Z_{m_i} \in \mathbb{R}^{l_i \times m_i}$, $l_i \le m_i$, are given cutting matrices. The generating function z_d of the Toeplitz matrix is a trigonometric polynomial, and should be

carefully chosen based on the information we know from the symbol of the matrix A (with size $m = m_1 \cdots m_d$) of the linear system to be solved. We refer to [34, Section 7.2] and [18, Section 6] for precise constraints on z_d, such that the corresponding projector possesses good approximation properties.

In our IgA setting, we focus on two particular projectors $P_{\boldsymbol{m}}$ and $Q_{\boldsymbol{m}}$. For any odd $m \geq 3$ (resp. for any m multiple of 3) let us denote by U_m (resp. V_m) the cutting matrix of size $\frac{m-1}{2} \times m$ (resp. $\frac{m}{3} \times m$) given by

$$U_m := \begin{bmatrix} 0 & 1 & & & & & & 0 \\ & & 0 & 1 & & & & 0 \\ & & & & \ddots & & & \vdots \\ & & & & & 0 & 1 & 0 \end{bmatrix} \in \mathbb{R}^{\frac{m-1}{2} \times m},$$

$$(\text{resp. } V_m := \begin{bmatrix} 1 & 0 & 0 & & & & & & \\ & & & 1 & 0 & 0 & & & \\ & & & & & & \ddots & & \\ & & & & & & & 1 & 0 & 0 \end{bmatrix} \in \mathbb{R}^{\frac{m}{3} \times m}).$$

For any $\boldsymbol{m} \in \mathbb{N}^d$ with odd $m_1, \ldots, m_d \geq 3$ (resp. for any $\boldsymbol{m} \in \mathbb{N}^d$ with $m_1, \ldots, m_d$ multiple of 3), we define $U_{\boldsymbol{m}} := U_{m_1} \otimes \cdots \otimes U_{m_d}$ (resp. $V_{\boldsymbol{m}} := V_{m_1} \otimes \cdots \otimes V_{m_d}$). Then, we set

$$P_{\boldsymbol{m}} := U_{\boldsymbol{m}} \cdot T_{\boldsymbol{m}}(q_d), \qquad q_d(\theta_1, \ldots, \theta_d) := \prod_{j=1}^{d} (1 + \cos\theta_j), \tag{5.8}$$

$$Q_{\boldsymbol{m}} := V_{\boldsymbol{m}} \cdot T_{\boldsymbol{m}}(r_d), \qquad r_d(\theta_1, \ldots, \theta_d) := \prod_{j=1}^{d} (3 + 4\cos(2\theta_j) + 2\cos(4\theta_j)). \tag{5.9}$$

It can be shown that $P_{\boldsymbol{m}}$ and $Q_{\boldsymbol{m}}$ admit 'recursive expressions':

$$P_{\boldsymbol{m}} = \bigotimes_{j=1}^{d} P_{m_j}, \qquad P_{m_j} = \bigotimes_{j=1}^{d} U_{m_j} \cdot T_{m_j}(q), \qquad q(\theta) = 1 + \cos\theta,$$

$$Q_{\boldsymbol{m}} = \bigotimes_{j=1}^{d} Q_{m_j}, \qquad Q_{m_j} = \bigotimes_{j=1}^{d} V_{m_j} \cdot T_{m_j}(r), \qquad r(\theta) = 3 + 4\cos(2\theta) + 2\cos(4\theta).$$

We observe that

$$P_{\boldsymbol{m}} = \bigotimes_{j=1}^{d} \frac{1}{2} \underbrace{\begin{bmatrix} 1 & 2 & 1 & & & & & & \\ & & 1 & 2 & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & 2 & 1 \end{bmatrix}}_{m_j},$$

and that both $P_{\boldsymbol{m}}$ and $Q_{\boldsymbol{m}}$ have full rank $\prod_{j=1}^{d} \frac{m_j - 1}{2}$ and $\prod_{j=1}^{d} \frac{m_j}{3}$, being Kronecker products of full-rank matrices.

From the theory developed in [18, 34] we know that the first projector leads to a coarse-grid correction which is highly contractive in the subspace of low frequencies, those related to $\boldsymbol{\theta} := (\theta_1, \ldots, \theta_d)$ in a neighborhood of zero, whereas the second projector is highly contractive in the subspace of low frequencies and in a special subspace of the high frequency subspace

related to $\boldsymbol{\theta}$ in a neighborhood of any point $(\varepsilon_1, \ldots, \varepsilon_d)$ such that $\varepsilon_j \in \{0, \pi\}$ for all $j = 1, \ldots, d$.

Let us now consider $d = 1$ and our specific linear systems. The symbol associated to the sequence of coefficient matrices is $f_p(\theta)$, as described in Section 4.1. Because $\theta = 0$ is the only zero of the symbol and with order 2, we expect (see e.g. [1, 15]) that both the projectors with any classical smoother (Richardson, Gauss-Seidel, Conjugate Gradient) lead to two-grid, V-cycle, and W-cycle algorithms with a convergence rate independent of the matrix size. However, for large p a numerical zero occurs at $\theta = \pi$ and therefore, while the projector P_m leads to a convergence rate worsening with p, we expect that the second projector Q_m leads to a convergence speed independent of the matrix size and only mildly dependent on p. These theoretical forecasts are numerically confirmed in Section 6.

If $d = 2$ and we consider our specific linear systems, the situation is more complicated than for the case $d = 1$, because of specific analytic features of the symbol $f_{p_1,p_2}^{(\nu_1,\nu_2)}(\theta_1, \theta_2)$ associated to the sequence of coefficient matrices, see Section 4.2. Since $(\theta_1, \theta_2) = (0, 0)$ is the only zero of order 2 of the symbol, we know (see e.g. [1, 15]) that both the projectors with any classical smoother (Richardson, Gauss-Seidel, Conjugate Gradient) induce two-grid, V-cycle, and W-cycle algorithms with a convergence rate independent of the matrix size. However, for large p_1 and large p_2, infinitely (sic!) many numerical zeros occur at any (θ_1, π), (π, θ_2), $\theta_1, \theta_2 \in [0, \pi]$. Thus, as in the one-dimensional setting, the first projector (with any classical smoother) leads to a convergence rate worsening with $\boldsymbol{p} := (p_1, p_2)$. The second projector is designed for coping with the four types of zeros of the symbol $(0, 0)$, $(0, \pi)$, $(\pi, 0)$, (π, π). Yet, $f_{p_1,p_2}^{(\nu_1,\nu_2)}$ possesses infinitely many numerical zeros, and hence also the second choice leads to a $\boldsymbol{p}$-dependent convergence speed. Furthermore, looking at the d-dimensional case, $d \geq 2$, the set of zeros grows because it is characterized by d facets (of dimension $d-1$) of the cube $[0, \pi]^d$, see equation (4.6).

Unfortunately, the situation is even worse: following the analysis in [18], it can be seen that there is no reduction strategy that can deal with infinitely many zeros along a line parallel to one of the axes. Therefore, any multigrid algorithm of that kind will show a $\boldsymbol{p}$-dependent convergence rate when $d \geq 2$.

5.5. PCG with $\boldsymbol{p}$-independent convergence rate

Let us start with recalling the PCG method (see e.g. [3]) for solving the linear system $A\mathbf{u} = \mathbf{b}$ with A an SPD matrix. Since we consider the preconditioned version of the CG method, we assume to have a matrix M such that M^{-1} is an approximation to A^{-1} and such that the product of M^{-1} and an arbitrary vector is easy to compute.

Algorithm 5.3. *Let $\mathbf{u}^{(k)}$ be a given approximation to the solution $\mathbf{u} = A^{-1}\mathbf{b}$ with A a real SPD matrix, and let M^{-1} be an approximation to A^{-1}. Then, the new approximation $\mathbf{u}^{(k+1)}$ is obtained as follows:*

1. *compute the approximation: $\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \alpha^{(k)}\mathbf{p}^{(k)}$, with the optimal step length $\alpha^{(k)} = (\mathbf{r}^{(k)\,T}\mathbf{z}^{(k)})/(\mathbf{p}^{(k)\,T}A\mathbf{p}^{(k)})$;*
2. *compute the residual: $\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} - \alpha^{(k)}A\mathbf{p}^{(k)}$;*
3. *compute the preconditioned residual: $\mathbf{z}^{(k+1)} = M^{-1}\mathbf{r}^{(k+1)}$;*
4. *compute the A-conjugate search direction: $\mathbf{p}^{(k+1)} = \mathbf{z}^{(k+1)} + \beta^{(k)}\mathbf{p}^{(k)}$, with $\beta^{(k)} = (\mathbf{z}^{(k+1)\,T}\mathbf{r}^{(k+1)})/(\mathbf{z}^{(k)\,T}\mathbf{r}^{(k)})$.*

If the vectors $\mathbf{r}^{(k)}$, $\mathbf{z}^{(k)}$, $\mathbf{p}^{(k)}$ are not yet computed by the algorithm in a previous step, then we initialize them as $\mathbf{r}^{(k)} = \mathbf{b} - A\mathbf{u}^{(k)}$, $\mathbf{z}^{(k)} = M^{-1}\mathbf{r}^{(k)}$, $\mathbf{p}^{(k)} = \mathbf{r}^{(k)}$.

Let us assume that $\boldsymbol{\beta} = \mathbf{0}$ in our model problem (2.1). Under this assumption, we know that $\frac{1}{n}A_n^{[p]}$ and $A_{\nu_1 n,\nu_2 n}^{[p_1,p_2]}$ are SPD (see Remark 2.1), and the PCG method can be applied to them. If $\boldsymbol{\beta} \neq \mathbf{0}$, we could replace the PCG method with the P-GMRES method, but this case will not be considered here.

We now focus on the construction of a preconditioner such that the PCG method will be $\boldsymbol{p}$-independent. The idea of a $\boldsymbol{p}$-independent PCG method has its theoretical foundation in the spectral results, concerning Toeplitz systems with Toeplitz preconditioners [14, 33], and in the study of the specific symbol of our sequences carried out in Section 4.

Let g_1 be a nonnegative, not identically zero and Lebesgue integrable function; then $T_{\boldsymbol{m}}(g_1)$ is a positive definite d-level Toeplitz matrix. Moreover, let g_2 be a real-valued and Lebesgue integrable function, such that g_2/g_1 is not constant. By following [14, 33], we know that all the eigenvalues of $T_{\boldsymbol{m}}^{-1}(g_1)T_{\boldsymbol{m}}(g_2)$ belong to the open set (r, R) with $r = \operatorname{ess\,inf} g_2/g_1$, $R = \operatorname{ess\,sup} g_2/g_1$ and

$$\left\{T_{\boldsymbol{m}}^{-1}(g_1)T_{\boldsymbol{m}}(g_2)\right\} \sim_\lambda g_2/g_1.$$

For $d = 1$ and $\{\frac{1}{n}A_n^{[p]}\}$, the symbol is $f_p(\theta) = h_{p-1}(\theta)(2 - 2\cos(\theta))$. This implies

$$\{T_{n+p-2}^{-1}(h_{p-1})\frac{1}{n}A_n^{[p]}\} \sim_\lambda f_p/h_{p-1} = 2 - 2\cos(\theta),$$

which is the symbol of the standard FD approximation given in (5.7) for $d = 1$, and the symbol $f_p/h_{p-1} = 2 - 2\cos(\theta)$ is indeed p-independent. Hence, if we take the PCG method with $T_{n+p-2}(h_{p-1})$ as preconditioner, we have a p-independent method. Unfortunately, it is slowly convergent when the matrix size increases (see Table 6 for a numerical example).

However, in view of the multi-iterative approach, we can build a totally robust method as follows: we consider a basic coarse-grid operator working in the low frequencies (like in the case of a standard FD approximation), and we include the PCG method, with preconditioner $T_{n+p-2}(h_{p-1})$, in the smoothing strategy. Thus, the coarse-grid operator will be responsible for the optimality of the method (a convergence speed independent of the matrix size) and the chosen smoother will bring the p-independence, taking care of the numerical zero at π for large p. In conclusion, the global multi-iterative method will be totally robust with respect to n and p, while the standard coarse-grid correction alone is not convergent and the PCG method alone is p-independent, but slowly convergent when the matrix size increases (see Section 6 for some numerical illustrations).

The good news is that the above technique can be generalized to any dimensionality (unlike the size reduction strategy which works only for $d = 1$).

Indeed, for $d = 2$, thanks to Lemma 4.1 (item 1), the symbol $f_{p_1,p_2}^{(\nu_1,\nu_2)}$ can be factored as follows:

$$f_{p_1,p_2}^{(\nu_1,\nu_2)}(\theta_1,\theta_2) = h_{p_2-1}(\theta_1)h_{p_1-1}(\theta_2)\left[\frac{\nu_2}{\nu_1}w_{p_1}(\theta_2)(2-2\cos\theta_1) + \frac{\nu_1}{\nu_2}w_{p_2}(\theta_1)(2-2\cos\theta_2)\right],$$

where $w_p(\theta) := \frac{h_p(\theta)}{h_{p-1}(\theta)}$ is a function well-separated from zero, uniformly with respect to $\theta \in [0,\pi]$ and with respect to $p \geq 1$. This means that the function between square brackets does not have numerical zeros and only has a zero at $\boldsymbol{\theta} = \mathbf{0}$. This zero does not create problems to our two-grid schemes, because the standard projector given in (5.8) takes care of it. Therefore, the function $h_{p_2-1}(\theta_1)h_{p_1-1}(\theta_2)$ is responsible for the existence of numerical zeros at the edge points (4.4) when p_1, p_2 are large. In other words, the same function is responsible for the poor behavior of our two-grid and multigrid schemes, with any classical smoother, and when p_1, p_2 are large. Consequently, we consider a preconditioner of the form

$$T_{\nu_2 n+p_2-2,\nu_1 n+p_1-2}(h_{p_2-1} \otimes h_{p_1-1}) = T_{\nu_2 n+p_2-2}(h_{p_2-1}) \otimes T_{\nu_1 n+p_1-2}(h_{p_1-1}). \tag{5.10}$$

The choice of using a PCG method with preconditioner (5.10) as a smoother is made in order to 'erase' all the numerical zeros at the edge points (4.4).

In addition, the proposed preconditioner (5.10) is effectively solvable: due to the tensor-product structure, the computational cost for solving a linear system with the preconditioning matrix in (5.10) is linear in the matrix size $(\nu_2 n + p_2 - 2)(\nu_1 n + p_1 - 2)$. More precisely, by the properties of the Kronecker product it holds

$$T^{-1}_{\nu_2 n+p_2-2,\nu_1 n+p_1-2}(h_{p_2-1} \otimes h_{p_1-1}) = T^{-1}_{\nu_2 n+p_2-2}(h_{p_2-1}) \otimes T^{-1}_{\nu_1 n+p_1-2}(h_{p_1-1}).$$

Let $\mathbf{b} \in \mathbb{R}^{(\nu_1 n+p_1-2)(\nu_2 n+p_2-2)}$ be the vector obtained by stacking the columns of the matrix $B \in \mathbb{R}^{(\nu_1 n+p_1-2)\times(\nu_2 n+p_2-2)}$, and define the corresponding stacking operator by

$$\mathbf{b} = \mathrm{vec}(B).$$

Then, the linear system $T_{\nu_2 n+p_2-2,\nu_1 n+p_1-2}(h_{p_2-1} \otimes h_{p_1-1})\mathbf{u} = \mathbf{b}$ can be solved by

$$\mathbf{u} = \mathrm{vec}\left(T_{\nu_1 n+p_1-2}(h_{p_1-1})^{-1}\, B\, T_{\nu_2 n+p_2-2}(h_{p_2-1})^{-T}\right).$$

This requires to solve $(\nu_2 n + p_2 - 2)$ linear systems with the banded Toeplitz matrix $T_{\nu_1 n+p_1-2}(h_{p_1-1})$, plus $(\nu_1 n+p_1-2)$ linear systems with $T_{\nu_2 n+p_2-2}(h_{p_2-1})^T$, see [27, Lemma 4.3.1]. Of course, this trick does not apply to the original system, which consists of sums of tensor-product matrices.

Finally, we can easily generalize the above results to the d-dimensional setting. From Section 4.3, see (4.5), it follows that

$$T_{\boldsymbol{m}}\left(\prod_{j=1}^{d} h_{p_j-1}(\theta_j)\right), \qquad m_j = \nu_j n + p_j - 2, \tag{5.11}$$

is a preconditioner with a $\boldsymbol{p}$-independent convergence rate for our coefficient matrices in the d-dimensional case (see Tables 6 and 12 for $d = 1, 2$). Therefore, in the spirit of the multi-iterative approach, we consider a basic coarse-grid operator working in the low frequencies (for a symbol like (5.7)) and we include the PCG with preconditioner (5.11) in the smoothing strategy. In this way, the coarse-grid operator will be responsible for the optimality of the method (a convergence speed independent of the matrix size) and the chosen smoother will induce the $\boldsymbol{p}$-independence, taking care of the numerical zeros in (4.6). In conclusion, the global multi-iterative method will be totally robust with respect to $\boldsymbol{n}$ and $\boldsymbol{p}$ (and surprisingly enough it seems also with respect to the dimensionality d).

Remark 5.1. *From the discussion above (in the 1D case), one could guess that the PCG method with $T_{n+p-2}(f_p)$ as preconditioner is substantially robust both with respect to n and p, because*

$$\{T^{-1}_{n+p-2}(f_p)\frac{1}{n}A^{[p]}_n\} \sim_\lambda f_p/f_p = 1.$$

This is numerically illustrated in Table 7. Unfortunately, this naive choice is not so practical. Solving a linear system with $T_{n+p-2}(f_p)$ is almost as expensive as solving the original system. And more importantly, the choice $T_{n+p-2}(f_p)$ cannot be effectively generalized to the higher dimensional setting. For example, in the 2D case, the PCG method with $T_{\nu_2 n+p_2-2}(f_{p_2}) \otimes T_{\nu_1 n+p_1-2}(f_{p_1})$ as preconditioner does not work (see Table 13). The explanation is clear: the function $f_{p_2} \otimes f_{p_1}$ and the symbol of our matrices $f^{(\nu_1,\nu_2)}_{p_1,p_2}$ possess two sets of zeros with a completely different structure. On the other hand, the use of $T_{\nu_2 n+p_2-2,\nu_1 n+p_1-2}(f^{(\nu_1,\nu_2)}_{p_1,p_2})$ as a

possible preconditioner is also twofold unsuccessful. First, its cost is prohibitive because of the lack of the tensor-product structure of the preconditioner. Second, it is not so effective because of the increasing number of outliers, due to the rank of $A^{[p_1,p_2]}_{\nu_1 n,\nu_2 n} - T_{\nu_2 n+p_2-2,\nu_1 n+p_1-2}(f^{(\nu_1,\nu_2)}_{p_1,p_2})$*, which is proportional to* n*, with a multiplicative constant depending on* ν_1,ν_2,p_1,p_2*.*

6. Algorithms and their performances: 1D

We start with a careful testing of two-grid methods with the classical projector (5.8) and with several smoothers. Note that the V-cycle and W-cycle convergence cannot be better than the one of the corresponding two-grid method. Then, we proceed with a different size reduction strategy and with the full multi-iterative approach sketched in Section 5.5.

6.1. Two-grid methods

We now illustrate the performance of two-grid methods with the classical projector $P_n^{[p]} := P_{n+p-2}$ given in (5.8), which induces a coarse-grid correction effective in the low frequencies. We only consider two-grid methods without pre-smoothing steps and with a single post-smoothing step.

Table 2 shows the results of some numerical experiments for $TG(\overline{S}_n^{[p]}, P_n^{[p]})$, with $\overline{S}_n^{[p]}$ being the iteration matrix of the relaxed Richardson method with parameter ω, see (5.3). We fixed $\beta = \gamma = 0$, so that $\frac{1}{n}A_n^{[p]} = K_n^{[p]}$ and $\overline{S}_n^{[p]} = I - \omega K_n^{[p]}$. Then, for $p = 1,\dots,6$ we determined experimentally the best Richardson parameter $\omega^{[p]}$, in the sense that $\omega^{[p]}$ minimizes $\overline{\rho}_n^{[p]} := \rho(TG(\overline{S}_n^{[p]}, P_n^{[p]}))$ with $n = 2560$ (if p is odd) and $n = 2561$ (if p is even) among all $\omega \in \mathbb{R}$ with at most four nonzero decimal digits after the comma. We note that the choice $\omega^{[1]} = 1/3$ has a theoretical motivation, because it imposes a fast convergence both in high and intermediate frequencies. Finally, we computed the spectral radii $\overline{\rho}_n^{[p]}$ for increasing values of n.

In all the considered experiments, the proposed two-grid scheme is optimal. Moreover, as $n \to \infty$, $\overline{\rho}_n^{[p]}$ converges to a limit $\overline{\rho}_\infty^{[p]}$, which is minimal not for $p = 1$ but for $p = 2$. A theoretical explanation of this phenomenon is given in [16]. When p increases from 2 to 6, we observe that $\overline{\rho}_\infty^{[p]}$ increases as well. In view of the theoretical interpretation based on the symbol f_p given in [16], $\overline{\rho}_\infty^{[p]}$ is expected to converge exponentially to 1 as $p \to \infty$, and in fact, even for moderate values of p such as $p = 5, 6$, we see from Table 2 that the value $\overline{\rho}_\infty^{[p]}$ is not satisfactory. This 'exponentially poor' behavior can be related to the fact that $f_p(\pi)/M_{f_p}$ exponentially approaches 0 when p increases (see Figure 1 and Table 1). Finally, from some numerical experiments we observe that $\rho(K_n^{[4]}) \approx 1.8372$, $\forall n \geq 15$. Therefore, for the set of indices $\mathcal{I}_4 = \{81, 161, \dots, 2561\}$ considered in Table 2, the best parameter $\omega^{[4]} = 1.2229$ produces a non-convergent smoother $\overline{S}_n^{[4]} = I - 1.2229\, K_n^{[4]}$ having $\rho(\overline{S}_n^{[4]}) \approx 1.2467$. This shows that the two-grid scheme can be convergent even when the smoother $\overline{S}_n^{[p]}$ is not and, moreover, $\overline{\rho}_n^{[p]}$ can attain its minimum at a value of $\omega^{[p]}$ for which $\rho(\overline{S}_n^{[p]}) > 1$, according to the multi-iterative idea (see Section 5 and [30]).

Table 3 illustrates the behavior of $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$ in the case $\beta = \gamma = 0$, for $p = 1,\dots,6$, with $\widehat{S}_n^{[p]}$ being the iteration matrix of the relaxed Gauss-Seidel method, see (5.4). Like in Table 2, the relaxation parameter $\omega^{[p]}$ was chosen so as to minimize $\widehat{\rho}_n^{[p]} := \rho(TG(\widehat{S}_n^{[p]}, P_n^{[p]}))$ with $n = 2560$ (if p is odd) and $n = 2561$ (if p is even) among all $\omega \in \mathbb{R}$ with at most four nonzero decimal digits after the comma. It follows from Table 3 that, except for the particular case $p = 2$, the use of the Gauss-Seidel smoother improves the convergence rate

n	$\overline{\rho}_n^{[1]}$ $[\omega^{[1]} = 1/3]$	$\overline{\rho}_n^{[3]}$ $[\omega^{[3]} = 1.0368]$	$\overline{\rho}_n^{[5]}$ $[\omega^{[5]} = 1.2576]$
80	0.3333333	0.4479733	0.8927544
160	0.3333333	0.4474586	0.8926293
320	0.3333333	0.4472015	0.8925948
640	0.3333333	0.4470729	0.8925948
1280	0.3333333	0.4470366	0.8925948
2560	0.3333333	0.4470391	0.8925948
n	$\overline{\rho}_n^{[2]}$ $[\omega^{[2]} = 0.7311]$	$\overline{\rho}_n^{[4]}$ $[\omega^{[4]} = 1.2229]$	$\overline{\rho}_n^{[6]}$ $[\omega^{[6]} = 1.2235]$
81	0.0257459	0.7373412	0.9596516
161	0.0254342	0.7371979	0.9595077
321	0.0252866	0.7371256	0.9594351
641	0.0252153	0.7371016	0.9593993
1281	0.0252000	0.7371016	0.9593993
2561	0.0252000	0.7371016	0.9593993

Table 2: Values of $\overline{\rho}_n^{[p]} := \rho(TG(\overline{S}_n^{[p]}, P_n^{[p]}))$ in the case $\beta = \gamma = 0$, for the specified parameter $\omega^{[p]}$.

n	$\widehat{\rho}_n^{[1]}$ $[\omega^{[1]} = 0.9065]$	$\widehat{\rho}_n^{[3]}$ $[\omega^{[3]} = 0.9483]$	$\widehat{\rho}_n^{[5]}$ $[\omega^{[5]} = 1.1999]$
80	0.1762977	0.1486937	0.4279346
160	0.1771878	0.1534242	0.4491173
320	0.1956301	0.1567792	0.4628558
640	0.2228058	0.1589204	0.4710180
1280	0.2358223	0.1602392	0.4758293
2560	0.2416926	0.1609750	0.4786945
n	$\widehat{\rho}_n^{[2]}$ $[\omega^{[2]} = 0.9109]$	$\widehat{\rho}_n^{[4]}$ $[\omega^{[4]} = 1.0602]$	$\widehat{\rho}_n^{[6]}$ $[\omega^{[6]} = 1.3292]$
81	0.0648736	0.2972510	0.5631940
161	0.0648736	0.3110761	0.5852798
321	0.0648736	0.3201033	0.6002364
641	0.0648736	0.3255332	0.6104147
1281	0.0648736	0.3286511	0.6164439
2561	0.0649656	0.3304592	0.6197837

Table 3: Values of $\widehat{\rho}_n^{[p]} := \rho(TG(\widehat{S}_n^{[p]}, P_n^{[p]}))$ in the case $\beta = \gamma = 0$, for the specified parameter $\omega^{[p]}$.

of the two-grid. However, we also observe that $\widehat{\rho}_n^{[p]}$ presents the same dependence on p as $\overline{\rho}_n^{[p]}$: the scheme is optimal, but its asymptotic convergence rate attains its minimum for $p = 2$ and then worsens as p increases from 2 to 6. As explained in Sections 4.1 and 5.4, we know that such a worsening is an intrinsic feature of the problem and is related to the fact that $f_p(\pi)/M_{f_p}$ converges exponentially to 0 for increasing p. In other words, the symbol f_p shows a numerical zero at π, inducing an ill-conditioning in the high frequencies, where our coarse-grid operator is not effective.

We now compare $TG(\overline{S}_n^{[p]}, P_n^{[p]})$ and $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$ on the linear system $K_n^{[p]}\mathbf{u} = \mathbf{b}$, coming from the B-spline IgA approximation of the model problem (2.5) with $\beta = \gamma = 0$ and $\mathrm{f} = 1$. In Table 4, the considered linear system was solved for $p = 1, \ldots, 6$ and for increasing values of n by means of $TG(\overline{S}_n^{[p]}, P_n^{[p]})$ (with $\omega^{[p]}$ as in Table 2) and $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$ (with $\omega^{[p]}$ as in Table 3). For each pair (p, n), $\overline{c}_n^{[p]}$ and $\widehat{c}_n^{[p]}$ are, respectively, the minimal number of iteration steps needed by $TG(\overline{S}_n^{[p]}, P_n^{[p]})$ and $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$, both started with initial guess $\mathbf{u}^{(0)} = \mathbf{0}$,

n	$\overline{c}_n^{[1]}[1/3]$	$\widehat{c}_n^{[1]}[0.9065]$	$\overline{c}_n^{[3]}[1.0368]$	$\widehat{c}_n^{[3]}[0.9483]$	$\overline{c}_n^{[5]}[1.2576]$	$\widehat{c}_n^{[5]}[1.1999]$
80	17	14	24	11	162	24
160	17	14	24	11	165	24
320	17	14	25	11	168	25
640	17	14	25	11	171	25
1280	17	14	26	11	174	26
2560	17	14	26	11	177	26
n	$\overline{c}_n^{[2]}[0.7311]$	$\widehat{c}_n^{[2]}[0.9109]$	$\overline{c}_n^{[4]}[1.2229]$	$\widehat{c}_n^{[4]}[1.0602]$	$\overline{c}_n^{[6]}[1.2235]$	$\widehat{c}_n^{[6]}[1.3292]$
81	6	8	61	16	448	34
161	6	8	62	17	456	35
321	6	8	63	17	464	36
641	6	8	64	17	472	36
1281	6	8	65	18	481	37
2561	6	8	66	18	489	38

Table 4: Number of iteration steps $\overline{c}_n^{[p]}$ and $\widehat{c}_n^{[p]}$ needed by $TG(\overline{S}_n^{[p]}, P_n^{[p]})$ and $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$ respectively, for solving $K_n^{[p]}\mathbf{u} = \mathbf{b}$ up to a precision of 10^{-8}. The methods have been started with $\mathbf{u}^{(0)} = \mathbf{0}$. The parameter $\omega^{[p]}$ is specified between the brackets $[\cdot]$.

to compute a vector $\mathbf{u}^{(c)}$ whose relative error in the 2-norm is less than 10^{-8}, i.e.,

$$\|\mathbf{b} - K_n^{[p]}\mathbf{u}^{(c)}\| \leq 10^{-8}\|\mathbf{b}\|. \tag{6.1}$$

6.2. A few proposals for improving the two-grid convergence rate

Despite their optimality, the basic two-grid schemes $TG(\overline{S}_n^{[p]}, P_n^{[p]})$ and $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$ suffer from the very same 'pathology', because – as already discussed – their convergence rate rapidly worsens when p increases. However, we can say that the global number of iterations is acceptable with the Gauss-Seidel smoothing in Table 4.

To overcome this pathological problem, in Section 6.2.1 we follow the idea in [18] and we design a couple of two-grid methods that use a different size reduction, i.e. with the projector $Q_n^{[p]} := Q_{n+p-2}$ given in (5.9). This projector is characterized by a reduction factor 3 and, as explained in Section 5.4, leads to a coarse-grid operator which is effective both in high and low frequencies.

In Section 6.2.2, following the multi-iterative idea sketched in Section 5.5, we replace and test, in the two-grid Algorithm 5.1, the smoothers $\overline{S}_n^{[p]}$ and $\widehat{S}_n^{[p]}$ with a proper PCG method, whose preconditioner takes care of dampening the 'high frequencies' corresponding to values of θ near π. With such a PCG as smoother, we can keep on using the projector $P_n^{[p]}$ working in the low frequency space.

6.2.1. A different size reduction

The two-grid methods $TG(\overline{S}_n^{[p]}, Q_n^{[p]})$ and $TG(\widehat{S}_n^{[p]}, Q_n^{[p]})$ are designed for the matrix $\frac{1}{n}A_n^{[p]}$, with $n \in \{n \geq 2 :\ n+p-2 \text{ multiple of } 3\}$. They use the same smoothers $\overline{S}_n^{[p]}$ and $\widehat{S}_n^{[p]}$ as before, but the projector is now $Q_n^{[p]} := Q_{n+p-2}$, as defined in (5.9) for $d = 1$ and $\boldsymbol{m} = n + p - 2$. In this way, $TG(\overline{S}_n^{[p]}, Q_n^{[p]})$ and $TG(\widehat{S}_n^{[p]}, Q_n^{[p]})$ adopt a reduction strategy with reduction factor 3 instead of 2.

The numerical experiments shown in Table 5 are completely analogous to those in Tables 2–3. The chosen $\omega^{[p]}$ is the minimizer, among all possible $\omega \in \mathbb{R}$ with at most four nonzero

n	$\overline{\varrho}_n^{[1]}$ [0.5714]	$\overline{\varrho}_n^{[4]}$ [1.5004]	$\widehat{\varrho}_n^{[1]}$ [1.2690]	$\widehat{\varrho}_n^{[4]}$ [0.9937]
73	0.7397897	0.5739123	0.6729043	0.3405459
145	0.7414768	0.5713995	0.6816279	0.3742356
289	0.7420542	0.5701637	0.6902198	0.3952195
577	0.7422413	0.5695509	0.6957753	0.4077374
1153	0.7422986	0.5692458	0.6991102	0.4150699
2305	0.7423146	0.5690231	0.7010014	0.4191964
n	$\overline{\varrho}_n^{[2]}$ [1.0497]	$\overline{\varrho}_n^{[5]}$ [1.6414]	$\widehat{\varrho}_n^{[2]}$ [1.1305]	$\widehat{\varrho}_n^{[5]}$ [1.0585]
72	0.5742882	0.6718405	0.5230691	0.3674381
144	0.5744819	0.6710837	0.5391576	0.3839339
288	0.5745326	0.6707105	0.5516503	0.3964006
576	0.5745456	0.6705251	0.5590383	0.4046942
1152	0.5746318	0.6704328	0.5633724	0.4097680
2304	0.5746582	0.6703890	0.5658265	0.4127329
n	$\overline{\varrho}_n^{[3]}$ [1.2917]	$\overline{\varrho}_n^{[6]}$ [1.7544]	$\widehat{\varrho}_n^{[3]}$ [1.0400]	$\widehat{\varrho}_n^{[6]}$ [1.0941]
71	0.5504342	0.7844256	0.4217059	0.4955220
143	0.5433287	0.7840813	0.4472211	0.4987532
287	0.5399472	0.7839123	0.4631007	0.5085735
575	0.5382974	0.7838285	0.4726064	0.5172697
1151	0.5374824	0.7838252	0.4781502	0.5225642
2303	0.5371722	0.7838272	0.4812615	0.5256146

Table 5: Values of $\overline{\varrho}_n^{[p]} := \rho(TG(\overline{S}_n^{[p]}, Q_n^{[p]}))$ and $\widehat{\varrho}_n^{[p]} := \rho(TG(\widehat{S}_n^{[p]}, Q_n^{[p]}))$ in the case $\beta = \gamma = 0$, for the specified parameter $\omega^{[p]}$ placed between the brackets $[\cdot]$.

decimal digits after the comma, of $\overline{\varrho}_n^{[p]} := \rho(TG(\overline{S}_n^{[p]}, Q_n^{[p]}))$ and $\widehat{\varrho}_n^{[p]} := \rho(TG(\widehat{S}_n^{[p]}, Q_n^{[p]}))$ with n taken to be 2305, 2304 or 2303 depending on the choice of p. By comparing Table 5 with Tables 2–3, we observe that for small p the two-grid methods with the projector $P_n^{[p]}$ have a better convergence rate than their counterparts with the projector $Q_n^{[p]}$, but when p is large the opposite happens. Hence, the two-grid methods with the projector $Q_n^{[p]}$ perform better when $f_p(\pi)/M_{f_p}$ is small.

Finally, we remark that the computational cost of an iteration of $TG(\overline{S}_n^{[p]}, Q_n^{[p]})$ (resp. $TG(\widehat{S}_n^{[p]}, Q_n^{[p]})$) is less expensive than the computational cost of an iteration of $TG(\overline{S}_n^{[p]}, P_n^{[p]})$ (resp. $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$). Indeed, the system solved at the lower level has smaller size: one third of the size of the original system instead of one half, see [18], in particular [18, Eq. (5.4)], for some more details on the computational cost.

6.2.2. A multi-iterative method: two-grid with PCG as smoother

We first illustrate the PCG method (see Algorithm 5.3) applied to the linear system $K_n^{[p]}\mathbf{u} = \mathbf{b}$, coming from the B-spline IgA approximation of the model problem (2.5) with $\beta = \gamma = 0$ and $\mathrm{f} = 1$. Table 6 reports the number of iterations needed by the PCG method with preconditioner $T_{n+p-2}(h_{p-1})$ to compute a vector $\mathbf{u}^{(c)}$ satisfying a relative error less than 10^{-8}, see (6.1). We observe that the PCG method is p-independent, but slowly convergent when the matrix size increases. On the other hand, as shown in Table 7, the number of iterations needed by the PCG method with preconditioner $T_{n+p-2}(f_p)$ is independent of n and mildly depending on p, see Remark 5.1.

n	$c_n^{[1]}$	$c_n^{[2]}$	$c_n^{[3]}$	$c_n^{[4]}$	$c_n^{[5]}$	$c_n^{[6]}$
80	40	40	41	42	44	44
160	80	80	81	83	86	87
320	160	160	161	166	170	172
640	320	320	321	331	338	343
1280	640	640	641	658	672	683
2560	1280	1280	1281	1311	1337	1363

Table 6: The number of iterations $c_n^{[p]}$ needed by the PCG method with preconditioner $T_{n+p-2}(h_{p-1})$, for solving the system $K_n^{[p]}\mathbf{u} = \mathbf{b}$ up to a precision of 10^{-8}. The method has been started with $\mathbf{u}^{(0)} = \mathbf{0}$.

n	$c_n^{[1]}$	$c_n^{[2]}$	$c_n^{[3]}$	$c_n^{[4]}$	$c_n^{[5]}$	$c_n^{[6]}$
80	1	3	5	6	7	9
160	1	3	5	6	7	9
320	1	3	5	6	7	9
640	1	3	5	6	7	9
1280	1	3	5	6	7	9
2560	1	3	5	6	7	9

Table 7: The number of iterations $c_n^{[p]}$ needed by the PCG method with preconditioner $T_{n+p-2}(f_p)$, for solving the system $K_n^{[p]}\mathbf{u} = \mathbf{b}$ up to a precision of 10^{-8}. The method has been started with $\mathbf{u}^{(0)} = \mathbf{0}$.

As discussed in Section 5.5, the convergence rate of the two-grid method can be improved for large p by using the PCG method as smoother: we take a few PCG post-smoothing iterations (say $s^{[p]}$ iterations) with preconditioner $T_{n+p-2}(h_{p-1})$. Due to the presence of the PCG smoother, the resulting method is no more a stationary iterative method, and hence it is not a two-grid in the classical sense. However, using an expressive notation, we denote this method by $TG((PCG)^{s^{[p]}}, P_n^{[p]})$, where the exponent $s^{[p]}$ simply indicates that we apply $s^{[p]}$ steps of the PCG algorithm and it is assumed that the preconditioner is $T_{n+p-2}(h_{p-1})$.

Then, the same system $K_n^{[p]}\mathbf{u} = \mathbf{b}$ was solved for $p = 1, \ldots, 6$ and for increasing values of n by means of $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ and $TG((\widehat{S}_n^{[p]})^{s^{[p]}}, P_n^{[p]})$. The latter method, as indicated by the notation, is the same as $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$, except that now we apply $s^{[p]}$ smoothing iterations by $\widehat{S}_n^{[p]}$ instead of only one. This is done for making a fair comparison with $TG((PCG)^{s^{[p]}}, P_n^{[p]})$, in which $s^{[p]}$ steps of PCG are applied. For the smoother $\widehat{S}_n^{[p]}$ we used the same $\omega^{[p]}$ as in Table 3. Both $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ and $TG((\widehat{S}_n^{[p]})^{s^{[p]}}, P_n^{[p]})$ are started with $\mathbf{u}^{(0)} = \mathbf{0}$ and stopped at the first term $\mathbf{u}^{(c)}$ satisfying (6.1). The corresponding numbers of iterations are collected in Table 8.

We observe from Table 8 that $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ has a better performance than $TG((\widehat{S}_n^{[p]})^{s^{[p]}}, P_n^{[p]})$ not only for large p but also for small p, though the difference between the two methods is much more appreciable when p is large. In the 2D case, the difference in performance between their 2D variants is even more significant, see Section 7.2. Another observation from Table 8 is the following: provided we increase $s^{[p]}$ a little bit when p increases, the number of iterations $\widetilde{c}_n^{[p]}$ needed by $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ to reach the preassigned accuracy 10^{-8} is essentially independent of both n and p. This implies that $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ is robust not only with respect to n but also with respect to p.

n	$\widetilde{c}_n^{[1]}$ [2]	$\widehat{c}_n^{[1]}$ [0.9065]	$\widetilde{c}_n^{[3]}$ [2]	$\widehat{c}_n^{[3]}$ [0.9483]	$\widetilde{c}_n^{[5]}$ [3]	$\widehat{c}_n^{[5]}$ [1.1999]
80	4	7	6	6	5	8
160	3	7	6	6	5	8
320	3	7	6	6	5	9
640	3	7	6	6	6	9
1280	3	7	6	6	6	9
2560	3	7	6	6	6	9
n	$\widetilde{c}_n^{[2]}$ [2]	$\widehat{c}_n^{[2]}$ [0.9109]	$\widetilde{c}_n^{[4]}$ [3]	$\widehat{c}_n^{[4]}$ [1.0602]	$\widetilde{c}_n^{[6]}$ [3]	$\widehat{c}_n^{[6]}$ [1.3292]
81	6	7	5	6	6	12
161	6	7	5	6	6	12
321	6	7	5	6	6	12
641	7	7	5	6	6	12
1281	7	7	5	6	6	13
2561	7	8	6	6	6	13

Table 8: The number of iterations $\widetilde{c}_n^{[p]}$ and $\widehat{c}_n^{[p]}$ needed by $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ and $TG((\widehat{S}_n^{[p]})^{s^{[p]}}, P_n^{[p]})$ respectively, for solving $K_n^{[p]}\mathbf{u} = \mathbf{b}$ up to a precision of 10^{-8}. The methods have been started with $\mathbf{u}^{(0)} = \mathbf{0}$. The parameters $s^{[p]}$ and $\omega^{[p]}$ are specified between brackets $[\cdot]$ near the labels $\widetilde{c}_n^{[p]}$ and $\widehat{c}_n^{[p]}$, respectively.

n	$r_n^{[1]}$ $[s^{[1]} = 2]$	$r_n^{[3]}$ $[s^{[3]} = 2]$	$r_n^{[5]}$ $[s^{[5]} = 3]$
80	0.0013033	0.0136701	0.0102340
160	0.0005140	0.0128580	0.0061823
320	0.0002030	0.0113767	0.0064649
640	0.0000793	0.0083046	0.0066540
1280	0.0000580	0.0067475	0.0088666
2560	0.0001097	0.0128079	0.0092601
n	$r_n^{[2]}$ $[s^{[2]} = 2]$	$r_n^{[4]}$ $[s^{[4]} = 3]$	$r_n^{[6]}$ $[s^{[6]} = 3]$
81	0.0165580	0.0073166	0.0141648
161	0.0157647	0.0068267	0.0123459
321	0.0155298	0.0066495	0.0109502
641	0.0163428	0.0055003	0.0090523
1281	0.0173932	0.0043674	0.0090503
2561	0.0199358	0.0074949	0.0099375

Table 9: The mean error reduction factor $r_n^{[p]}$ given in (6.2), when the linear system $K_n^{[p]}\mathbf{u} = \mathbf{b}$ is solved up to a precision of 10^{-8} by means of $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ with the specified number $s^{[p]}$ of PCG smoothing iterations. The method has been started with $\mathbf{u}^{(0)} = \mathbf{0}$.

Finally, we look at the geometric mean of the error ratios in the 2-norm, i.e.,

$$\sqrt[c]{\frac{\|\mathbf{e}^{(c)}\|}{\|\mathbf{e}^{(c-1)}\|} \cdots \frac{\|\mathbf{e}^{(1)}\|}{\|\mathbf{e}^{(0)}\|}} = \sqrt[c]{\frac{\|\mathbf{e}^{(c)}\|}{\|\mathbf{e}^{(0)}\|}}, \tag{6.2}$$

where $\mathbf{e}^{(k)} := \mathbf{u} - \mathbf{u}^{(k)}$ is the error at step k and c is the stopping index. If $\mathbf{u}^{(k)}$ is computed by means of a stationary iterative method, then (6.2) is a good approximation of the spectral radius of the iteration matrix. Even though $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ is not a stationary method, this geometric mean can provide a similar performance measure as well. Table 9 reports these values, denoted by $r_n^{[p]}$ and computed for the same problems as described above with

$TG((PCG)^{s^{[p]}}, P_n^{[p]})$ as solver. We clearly see that the values $r_n^{[p]}$ are not only smaller than the corresponding values in Tables 2–3 (even when taking into account the number of smoothing iterations), but they also confirm that the proposed method is robust with respect to n and p.

Summarizing, $TG((PCG)^{s^{[p]}}, P_n^{[p]})$ is a totally robust method, not only with respect to n but also with respect to p. This property does not hold for $TG(\overline{S}_n^{[p]}, P_n^{[p]})$ and $TG(\widehat{S}_n^{[p]}, P_n^{[p]})$, because we have seen that both $\overline{\rho}_n^{[p]}$ and $\widehat{\rho}_n^{[p]}$ increase with p.

7. Algorithms and their performances: 2D

In this section we consider specialized two-grid methods for linear systems having $A_{n,n}^{[p,p]}$ as coefficient matrix. To this end, we are going to follow the recipe sketched in Section 5.2. In particular, we will exploit specific properties of the symbol $f_{p,p}^{(1,1)}$, see (4.3), in order to choose an appropriate projector.

7.1. Two-grid methods

We consider two-grid methods with the classical projector $P_{n,n}^{[p,p]} := P_{n+p-2,n+p-2}$ given in (5.8), which induces a coarse-grid correction effective in the low frequencies. Like in the 1D setting, we only consider two-grid methods without pre-smoothing steps and with a single post-smoothing step. We provide two choices of the smoother: the relaxed Richardson smoother with iteration matrix $\overline{S}_{n,n}^{[p,p]}$ and the relaxed Gauss-Seidel smoother with iteration matrix $\widehat{S}_{n,n}^{[p,p]}$, see (5.3)–(5.4). With the smoothers as above and the projector considered in (5.8), our two-grid procedure is defined completely for $A = A_{n,n}^{[p,p]}$, see Algorithm 5.1.

Table 10 shows the results of some numerical experiments in the case $\boldsymbol{\beta} = \mathbf{0}$, $\gamma = 0$. For $p = 1, \ldots, 6$, we determined experimentally the parameter $\omega^{[p,p]}$ minimizing the quantity $\overline{\rho}_{n,n}^{[p,p]} := \rho(TG(\overline{S}_{n,n}^{[p,p]}, P_{n,n}^{[p,p]}))$, where n is chosen to be 52 (if p is odd) or 53 (if p is even). Then, we computed the spectral radii $\overline{\rho}_{n,n}^{[p,p]}$ for increasing values of n. In all the considered experiments, the proposed two-grid method is optimal. However, for $p = 4, 5, 6$ the spectral radii are very close to 1, and this is not satisfactory for practical purposes. The numerical experiments in Table 11, obtained as those in Table 10, show a certain improvement in the two-grid convergence rate when using the relaxed Gauss-Seidel smoother instead of Richardson's. However, for large p, the values $\widehat{\rho}_{n,n}^{[p,p]}$ are still unsatisfactory.

7.2. A multi-iterative method: two-grid with PCG as smoother

The convergence rate of both the two-grid schemes $TG(\overline{S}_{n,n}^{[p,p]}, P_{n,n}^{[p,p]})$ and $TG(\widehat{S}_{n,n}^{[p,p]}, P_{n,n}^{[p,p]})$ rapidly worsens when p increases. Moreover, using a different size reduction, as we have done in Section 6.2.1 for the 1D case, does not work in the 2D case: the convergence rate is still poor for large p. The main reason, as explained in Section 5.4, is the presence of a large set of numerical zeros of the symbol $f_{p,p}^{(1,1)}$, see (4.4). Following the suggestion from Section 5.5, we now adopt a multi-iterative method, identical to the one tested in Section 6.2.2, which involves the PCG method as smoother.

Let us first illustrate the PCG method (see Algorithm 5.3) applied to the linear system $K_{n,n}^{[p,p]}\mathbf{u} = \mathbf{b}$, coming from the B-spline IgA approximation of the model problem (2.1) in the case $d = 2$ with $\Omega = (0,1)^2$, $\boldsymbol{\beta} = \mathbf{0}$, $\gamma = 0$ and $\mathrm{f} = 1$. Table 12 reports the number of iterations needed by the PCG method with preconditioner $T_{n+p-2}(h_{p-1}) \otimes T_{n+p-2}(h_{p-1})$ to compute a vector $\mathbf{u}^{(c)}$ satisfying a relative error less than 10^{-8}. As illustrated in Table 13, the PCG method with preconditioner $T_{n+p-2}(f_p) \otimes T_{n+p-2}(f_p)$ is not effective, since there

n	$\overline{\rho}_{n,n}^{[1,1]}$ $[\omega^{[1,1]} = 0.3335]$	$\overline{\rho}_{n,n}^{[3,3]}$ $[\omega^{[3,3]} = 1.3739]$	$\overline{\rho}_{n,n}^{[5,5]}$ $[\omega^{[5,5]} = 1.3293]$
16	0.3287279	0.9248227	0.9984590
28	0.3316020	0.9239241	0.9983433
40	0.3323146	0.9231361	0.9983185
52	0.3325944	0.9229755	0.9983134
n	$\overline{\rho}_{n,n}^{[2,2]}$ $[\omega^{[2,2]} = 1.1009]$	$\overline{\rho}_{n,n}^{[4,4]}$ $[\omega^{[4,4]} = 1.4000]$	$\overline{\rho}_{n,n}^{[6,6]}$ $[\omega^{[6,6]} = 1.2505]$
17	0.6085689	0.9885344	0.9997977
29	0.6085689	0.9881173	0.9997766
41	0.6085689	0.9880112	0.9997724
53	0.6085689	0.9879839	0.9997715

Table 10: Values of $\overline{\rho}_{n,n}^{[p,p]} := \rho(TG(\overline{S}_{n,n}^{[p,p]}, P_{n,n}^{[p,p]}))$ in the case $\boldsymbol{\beta} = \mathbf{0}$, $\gamma = 0$, for the specified parameter $\omega^{[p,p]}$.

n	$\widehat{\rho}_{n,n}^{[1,1]}$ $[\omega^{[1,1]} = 1.0035]$	$\widehat{\rho}_{n,n}^{[3,3]}$ $[\omega^{[3,3]} = 1.3143]$	$\widehat{\rho}_{n,n}^{[5,5]}$ $[\omega^{[5,5]} = 1.3990]$
16	0.1588106	0.6420608	0.9629505
28	0.1678248	0.6411764	0.9633667
40	0.1753106	0.6418579	0.9626834
52	0.1804148	0.6465563	0.9620579
n	$\widehat{\rho}_{n,n}^{[2,2]}$ $[\omega^{[2,2]} = 1.1695]$	$\widehat{\rho}_{n,n}^{[4,4]}$ $[\omega^{[4,4]} = 1.3248]$	$\widehat{\rho}_{n,n}^{[6,6]}$ $[\omega^{[6,6]} = 1.4914]$
17	0.2661407	0.8798035	0.9913084
29	0.2689991	0.8779954	0.9903263
41	0.2901481	0.8773914	0.9898795
53	0.3045791	0.8778602	0.9897372

Table 11: Values of $\widehat{\rho}_{n,n}^{[p,p]} := \rho(TG(\widehat{S}_{n,n}^{[p,p]}, P_{n,n}^{[p,p]}))$ in the case $\boldsymbol{\beta} = \mathbf{0}$, $\gamma = 0$, for the specified parameter $\omega^{[p,p]}$.

is an unsatisfactory dependency on n and p. We refer to Remark 5.1 for a brief explanation of this phenomenon.

Then, the same system $K_{n,n}^{[p,p]}\mathbf{u} = \mathbf{b}$ was solved for $p = 1, \ldots, 6$ and for increasing n, by means of $TG((PCG)^{s^{[p,p]}}, P_{n,n}^{[p,p]})$ and $TG((\widehat{S}_{n,n}^{[p,p]})^{s^{[p,p]}}, P_{n,n}^{[p,p]})$. The corresponding numbers of iterations steps are given in Table 14. For $\widehat{S}_{n,n}^{[p,p]}$ we used the same parameter $\omega^{[p,p]}$ as in Table 11. Both $TG((PCG)^{s^{[p,p]}}, P_{n,n}^{[p,p]})$ and $TG((\widehat{S}_{n,n}^{[p,p]})^{s^{[p,p]}}, P_{n,n}^{[p,p]})$ were started with $\mathbf{u}^{(0)} = \mathbf{0}$ and stopped at the first term $\mathbf{u}^{(c)}$ satisfying a criterion of relative error less than 10^{-8}. Table 15 reports the geometric mean (6.2) for $TG((PCG)^{s^{[p,p]}}, P_{n,n}^{[p,p]})$ as solver. Analogously to the 1D case (see Section 6.2.2), we can conclude that $TG((PCG)^{s^{[p,p]}}, P_{n,n}^{[p,p]})$ is totally robust, not only with respect to n but also with respect to p.

8. Multigrid with V-cycle and W-cycle

This section illustrates the numerical behavior of the V-cycle and W-cycle multigrid algorithms. Like for the two-grid algorithms, we observe an optimal convergence rate (see Tables 16–17). First, we consider again the linear systems $K_n^{[p]}\mathbf{u} = \mathbf{b}$ and $K_{n,n}^{[p,p]}\mathbf{u} = \mathbf{b}$, taking only the diffusion part of problem (2.1), $d = 1, 2$, discretized with B-splines and $\mathrm{f} = 1$. Afterwards, we briefly discuss the role of the advection term.

n	$c_{n,n}^{[1,1]}$	$c_{n,n}^{[2,2]}$	$c_{n,n}^{[3,3]}$	$c_{n,n}^{[4,4]}$	$c_{n,n}^{[5,5]}$	$c_{n,n}^{[6,6]}$
15	18	19	20	23	26	33
25	32	30	32	36	41	49
35	45	43	43	50	57	68
45	58	56	56	63	73	88
55	72	68	69	76	89	109

Table 12: The number of iterations $c_{n,n}^{[p,p]}$ needed by the PCG method with preconditioner $T_{n+p-2}(h_{p-1}) \otimes T_{n+p-2}(h_{p-1})$, for solving the system $K_{n,n}^{[p,p]}\mathbf{u} = \mathbf{b}$ up to a precision of 10^{-8}. The method has been started with $\mathbf{u}^{(0)} = \mathbf{0}$.

n	$c_{n,n}^{[1,1]}$	$c_{n,n}^{[2,2]}$	$c_{n,n}^{[3,3]}$	$c_{n,n}^{[4,4]}$	$c_{n,n}^{[5,5]}$	$c_{n,n}^{[6,6]}$
15	43	58	65	90	113	149
25	84	105	126	153	190	240
35	118	149	176	211	263	325
45	153	189	220	265	327	409
55	189	230	268	320	394	486

Table 13: The number of iterations $c_{n,n}^{[p,p]}$ needed by the PCG method with preconditioner $T_{n+p-2}(f_p) \otimes T_{n+p-2}(f_p)$, for solving the system $K_{n,n}^{[p,p]}\mathbf{u} = \mathbf{b}$ up to a precision of 10^{-8}. The method has been started with $\mathbf{u}^{(0)} = \mathbf{0}$.

n	$\widetilde{c}_{n,n}^{[1,1]}$ [2]	$\widehat{c}_{n,n}^{[1,1]}$ [1.0035]	$\widetilde{c}_{n,n}^{[3,3]}$ [2]	$\widehat{c}_{n,n}^{[3,3]}$ [1.3143]	$\widetilde{c}_{n,n}^{[5,5]}$ [4]	$\widehat{c}_{n,n}^{[5,5]}$ [1.3990]
16	6	7	6	16	7	69
28	6	7	6	15	6	59
40	6	7	6	14	6	54
52	6	7	6	14	6	51
64	6	7	6	14	6	48
76	6	7	6	14	6	46
n	$\widetilde{c}_{n,n}^{[2,2]}$ [2]	$\widehat{c}_{n,n}^{[2,2]}$ [1.1695]	$\widetilde{c}_{n,n}^{[4,4]}$ [3]	$\widehat{c}_{n,n}^{[4,4]}$ [1.3248]	$\widetilde{c}_{n,n}^{[6,6]}$ [6]	$\widehat{c}_{n,n}^{[6,6]}$ [1.4914]
17	6	8	6	33	6	157
29	6	8	6	30	6	127
41	6	8	6	29	6	115
53	6	8	6	28	5	108
65	6	9	6	27	5	102
77	6	9	6	27	5	98

Table 14: The number of iterations $\widetilde{c}_{n,n}^{[p,p]}$ and $\widehat{c}_{n,n}^{[p,p]}$ needed by $TG((PCG)^{s^{[p,p]}}, P_{n,n}^{[p,p]})$ and $TG((\widehat{S}_{n,n}^{[p,p]})^{s^{[p,p]}}, P_{n,n}^{[p,p]})$ respectively, for solving $K_{n,n}^{[p,p]}\mathbf{u} = \mathbf{b}$ up to a precision of 10^{-8}. The methods have been started with $\mathbf{u}^{(0)} = \mathbf{0}$. The parameters $s^{[p,p]}$ and $\omega^{[p,p]}$ are specified between brackets $[\cdot]$ near the labels $\widetilde{c}_{n,n}^{[p,p]}$ and $\widehat{c}_{n,n}^{[p,p]}$, respectively.

8.1. 1D case

Table 16 reports the numbers of iterations needed to solve the system $K_n^{[p]}\mathbf{u} = \mathbf{b}$ with the V-cycle and the W-cycle method, see Algorithm 5.2. We used the initial guess $\mathbf{u}^{(0)} = \mathbf{0}$ and the stopping criterion of the relative error less than 10^{-8}. We now explain in detail how our multigrid algorithms were constructed.

n	$r_{n,n}^{[1,1]}$ $[s^{[1,1]}=2]$	$r_{n,n}^{[3,3]}$ $[s^{[3,3]}=2]$	$r_{n,n}^{[5,5]}$ $[s^{[5,5]}=4]$
16	0.0264812	0.0319448	0.0594416
28	0.0203715	0.0231017	0.0398295
40	0.0172077	0.0200468	0.0316042
52	0.0151646	0.0194063	0.0260210
64	0.0137268	0.0186084	0.0249278
76	0.0126300	0.0173963	0.0247393
n	$r_{n,n}^{[2,2]}$ $[s^{[2,2]}=2]$	$r_{n,n}^{[4,4]}$ $[s^{[4,4]}=3]$	$r_{n,n}^{[6,6]}$ $[s^{[6,6]}=6]$
17	0.0286384	0.0326945	0.0546095
29	0.0239272	0.0271920	0.0436212
41	0.0195676	0.0240436	0.0360002
53	0.0193817	0.0212144	0.0240992
65	0.0171041	0.0190261	0.0216700
77	0.0164349	0.0174056	0.0196113

Table 15: The mean error reduction factor $r_{n,n}^{[p,p]}$ given in (6.2), when the linear system $K_{n,n}^{[p,p]}\mathbf{u}=\mathbf{b}$ is solved up to a precision of 10^{-8} by means of $TG((PCG)^{s^{[p,p]}}, P_{n,n}^{[p,p]})$ with the specified number $s^{[p,p]}$ of PCG smoothing iterations. The method has been started with $\mathbf{u}^{(0)}=\mathbf{0}$.

n	$\widetilde{c}_n^{[1]}[2]$	$\widehat{c}_n^{[1]}[0.9065]$	n	$\widetilde{c}_n^{[3]}[2]$	$\widehat{c}_n^{[3]}[0.9483]$	n	$\widetilde{c}_n^{[5]}[3]$	$\widehat{c}_n^{[5]}[1.1999]$
16	10 - 7	9 - 7	14	8 - 6	7 - 5	12	7 - 5	7 - 7
32	11 - 7	10 - 7	30	9 - 6	8 - 5	28	9 - 5	8 - 8
64	12 - 7	11 - 7	62	10 - 6	9 - 6	60	10 - 5	9 - 8
128	13 - 7	12 - 8	126	11 - 6	9 - 6	124	11 - 5	10 - 8
256	13 - 7	12 - 8	254	11 - 6	10 - 6	252	12 - 6	11 - 8
512	14 - 7	13 - 8	510	12 - 6	11 - 6	508	13 - 6	12 - 9
1024	14 - 7	14 - 8	1022	12 - 6	12 - 6	1020	13 - 6	13 - 9
n	$\widetilde{c}_n^{[2]}[2]$	$\widehat{c}_n^{[2]}[0.9109]$	n	$\widetilde{c}_n^{[4]}[3]$	$\widehat{c}_n^{[4]}[1.0602]$	n	$\widetilde{c}_n^{[6]}[3]$	$\widehat{c}_n^{[6]}[1.3292]$
15	8 - 6	7 - 6	13	8 - 6	6 - 5	11	7 - 5	10 - 10
31	10 - 6	9 - 7	29	9 - 6	8 - 6	27	9 - 6	12 - 12
63	11 - 6	10 - 7	61	10 - 6	9 - 6	59	9 - 6	12 - 12
127	11 - 6	11 - 7	125	11 - 6	10 - 6	123	11 - 6	12 - 12
255	12 - 7	11 - 7	253	12 - 6	11 - 6	251	12 - 6	12 - 12
511	13 - 7	12 - 7	509	12 - 6	12 - 6	507	13 - 6	13 - 12
1023	13 - 7	12 - 7	1021	13 - 6	13 - 6	1019	14 - 6	13 - 13

Table 16: The number of iterations $\widetilde{c}_n^{[p]}$ (resp. $\widehat{c}_n^{[p]}$) needed for solving $K_n^{[p]}\mathbf{u}=\mathbf{b}$ up to a precision of 10^{-8}, when using the multigrid cycle with $s^{[p]}$ smoothing steps by the PCG algorithm (resp. by the relaxed Gauss-Seidel smoother $\widehat{S}_{n,0}^{[p]}$) at the finest level, and one smoothing step by the simple Gauss-Seidel smoother $\widehat{S}_{n,i}^{[p]}$ at the coarse levels. The parameters $s^{[p]}$ and $\omega^{[p]}$ are specified between brackets $[\cdot]$ near the labels $\widetilde{c}_n^{[p]}$ and $\widehat{c}_n^{[p]}$, respectively. For each pair (p,n), the first entry in the cell corresponding to $\widetilde{c}_n^{[p]}$ refers to the V-cycle; the second entry to the W-cycle. The same holds for $\widehat{c}_n^{[p]}$.

The finest level is indicated by index 0, and the coarsest level by index $\ell_n^{[p]} := \log_2(n+p-1)-1$, assuming that $n+p-1$ is a power of 2. Let $K_{n,i}^{[p]}$ be the matrix at level i and its dimension is denoted by $m_i^{[p]}$, $0 \le i \le \ell_n^{[p]}$. In this notation, we have $K_{n,0}^{[p]} = K_n^{[p]}$,

$$K_{n,i+1}^{[p]} = P_{n,i}^{[p]} K_{n,i}^{[p]} (P_{n,i}^{[p]})^T, \quad i = 0, \ldots, \ell_n^{[p]} - 1,$$

and $K^{[p]}_{n,\ell^{[p]}_n}$ has dimension 1. In the above expression,

$$P^{[p]}_{n,i} := P_{m^{[p]}_i}, \quad i = 0, \ldots, \ell^{[p]}_n - 1,$$

is the projector at level i, defined by (5.8) for $d = 1$ and $\boldsymbol{m} = m^{[p]}_i$. Given the shape of $P_{m^{[p]}_i}$, one can show by induction on i that $m^{[p]}_{i+1} = (m^{[p]}_i - 1)/2$, $i = 0, \ldots, \ell^{[p]}_n - 1$, and $m^{[p]}_i = \frac{n+p-1}{2^i} - 1$, $i = 0, \ldots, \ell^{[p]}_n$.

We note that the choice of the projector $P^{[p]}_{n,i}$ at each level i has the same motivation as the projector $P^{[p]}_n$ for $K^{[p]}_n$, as discussed in Section 5.4. We know that $K^{[p]}_n$ has the symbol $f_{p,0} := f_p$. Then, referring to [34, Proposition 2.2] or [2, Proposition 2.5], it follows that $K^{[p]}_{n,i}$ has a symbol $f_{p,i}$ at level i sharing the same properties of the symbol $f_{p,0}$ at level 0: $f_{p,i}(0) = 0$, with $\theta = 0$ a zero of order two, and $f_{p,i}(\theta) > 0$ for all $\theta \in [-\pi, \pi]\backslash\{0\}$ (see also Section 3.7.1 in [36]). These properties coincide with those of f_p used in Section 5.4 for devising the appropriate projector $P^{[p]}_n$ for $K^{[p]}_n$.

Regarding the smoother, at each coarse level $i \geq 1$ we chose the standard Gauss-Seidel smoother in (5.4) without relaxation (i.e. $\omega = 1$). However, at the finest level $i = 0$ we considered two alternatives: $s^{[p]}$ smoothing iterations by the PCG method with preconditioner $T_{n+p-2}(h_{p-1})$, as in Section 6.2.2, or $s^{[p]}$ smoothing iterations by the relaxed Gauss-Seidel method $\widehat{S}^{[p]}_{n,0}$ with the relaxation parameter $\omega^{[p]}$ as in Table 3. Note that, due to the presence of the (optimal) parameter $\omega^{[p]}$, $\widehat{S}^{[p]}_{n,0}$ is different from $\widehat{S}^{[p]}_{n,i}$, $i \geq 1$.

At each level i, we first performed a coarse-grid correction, with one recursive call in the V-cycle and two recursive calls in the W-cycle, and then we applied one post-smoothing iteration by $\widehat{S}^{[p]}_{n,i}$ (if $i \geq 1$), or $s^{[p]}$ post-smoothing iterations by the PCG algorithm or $\widehat{S}^{[p]}_{n,0}$ (if $i = 0$). From Table 16 we can conclude that all the proposed multigrid methods have an optimal convergence rate. Moreover, the versions with a few PCG smoothing steps are totally robust, not only in n but also in p.

Finally, we want to motivate why the $s^{[p]}$ PCG smoothing steps were used only at the finest level. Let $M_{f_{p,i}} := \max_{\theta\in[-\pi,\pi]} f_{p,i}(\theta)$. Referring to [34, Proposition 2.2 (item 2)], and taking into account some additional numerical experiments that we performed, it seems that the numerical zero $\theta = \pi$ of $f_{p,0}/M_{f_{p,0}}$ disappears for $i \geq 1$, and each $f_{p,i}/M_{f_{p,i}}$, $i \geq 1$, only possesses the actual zero $\theta = 0$. Hence, a single smoothing iteration by the standard Gauss-Seidel method is all we need at the coarse levels $i \geq 1$.

8.2. 2D case

Table 17 reports the numbers of iterations needed to solve the system $K^{[p,p]}_{n,n}\mathbf{u} = \mathbf{b}$ with the V-cycle and the W-cycle method, see Algorithm 5.2. We used the same choice of initial guess and stopping criterion as for Table 16. The multigrid algorithms were constructed in a similar way as in the 1D case.

The finest level is again indicated by index 0, and the coarsest level by index $\ell^{[p]}_n := \log_2(n+p-1)-1$. Let $K^{[p,p]}_{n,n,i}$ be the matrix at level i, whose dimension is $(m^{[p]}_i)^2$, $0 \leq i \leq \ell^{[p]}_n$. We have

$$K^{[p,p]}_{n,n,i+1} = P^{[p,p]}_{n,n,i}\, K^{[p,p]}_{n,n,i}\, (P^{[p,p]}_{n,n,i})^T, \quad i = 0, \ldots, \ell^{[p]}_n - 1,$$

where

$$P^{[p,p]}_{n,n,i} := P_{m^{[p]}_i, m^{[p]}_i}, \quad i = 0, \ldots, \ell^{[p]}_n - 1,$$

is the projector at level i, defined by (5.8) for $d = 2$ and $\boldsymbol{m} = (m^{[p]}_i, m^{[p]}_i)$.

n	$\widetilde{c}_{n,n}^{[1,1]}$ [2]	$\widehat{c}_{n,n}^{[1,1]}$ [1.0035]	n	$\widetilde{c}_{n,n}^{[3,3]}$ [2]	$\widehat{c}_{n,n}^{[3,3]}$ [1.3143]	n	$\widetilde{c}_{n,n}^{[5,5]}$ [4]	$\widehat{c}_{n,n}^{[5,5]}$ [1.3990]
16	10 - 7	9 - 7	14	7 - 6	16 - 16	12	7 - 7	85 - 85
32	11 - 7	10 - 7	30	9 - 6	15 - 15	28	8 - 6	59 - 59
64	12 - 7	11 - 7	62	9 - 6	14 - 14	60	10 - 6	49 - 49
128	13 - 7	12 - 7	126	10 - 6	13 - 13	124	11 - 6	42 - 42
n	$\widetilde{c}_{n,n}^{[2,2]}$ [2]	$\widehat{c}_{n,n}^{[2,2]}$ [1.1695]	n	$\widetilde{c}_{n,n}^{[4,4]}$ [3]	$\widehat{c}_{n,n}^{[4,4]}$ [1.3248]	n	$\widetilde{c}_{n,n}^{[6,6]}$ [6]	$\widehat{c}_{n,n}^{[6,6]}$ [1.4914]
15	8 - 6	8 - 8	13	7 - 6	37 - 37	11	7 - 7	204 - 204
31	9 - 6	8 - 8	29	8 - 6	30 - 30	27	8 - 6	129 - 129
63	10 - 6	9 - 9	61	10 - 6	27 - 28	59	10 - 6	105 - 105
127	11 - 6	10 - 9	125	11 - 6	25 - 25	123	11 - 6	86 - 87

Table 17: The number of iterations $\widetilde{c}_{n,n}^{[p,p]}$ (resp. $\widehat{c}_{n,n}^{[p,p]}$) needed for solving $K_{n,n}^{[p,p]}\mathbf{u} = \mathbf{b}$ up to a precision of 10^{-8}, when using the multigrid cycle with $s^{[p,p]}$ smoothing steps by the PCG algorithm (resp. by the relaxed Gauss-Seidel smoother $\widehat{S}_{n,n,0}^{[p,p]}$) at the finest level and one smoothing step by the simple Gauss-Seidel smoother $\widehat{S}_{n,n,i}^{[p,p]}$ at the coarse levels. The parameters $s^{[p,p]}$ and $\omega^{[p,p]}$ are specified between brackets $[\cdot]$ near the labels $\widetilde{c}_{n,n}^{[p,p]}$ and $\widehat{c}_{n,n}^{[p,p]}$, respectively. For each pair (p,n), the first entry in the cell corresponding to $\widetilde{c}_{n,n}^{[p,p]}$ refers to the V-cycle; the second entry to the W-cycle. The same holds for $\widehat{c}_{n,n}^{[p,p]}$.

Regarding the smoother, we took the same choices as in the 1D case. At each coarse level $i \geq 1$ we used the standard Gauss-Seidel smoother without relaxation. However, at the finest level $i = 0$ we used either $s^{[p,p]}$ smoothing iterations by the PCG algorithm with preconditioner (5.10) or $s^{[p,p]}$ smoothing iterations by the relaxed Gauss-Seidel method $\widehat{S}_{n,n,0}^{[p,p]}$ with the relaxation parameter $\omega^{[p,p]}$ as in Table 11.

At each level i, we first performed a coarse-grid correction, with one recursive call in the V-cycle and two recursive calls in the W-cycle, and then we applied one post-smoothing iteration by $\widehat{S}_{n,n,i}^{[p,p]}$ (if $i \geq 1$), or $s^{[p,p]}$ post-smoothing iterations by the PCG algorithm or $\widehat{S}_{n,n,0}^{[p,p]}$ (if $i = 0$).

When using a few PCG smoothing steps at the finest level, we can conclude from Tables 16–17 that the resulting V-cycle and W-cycle multigrid algorithms have a convergence rate that is independent not only of n but also of p. This means that they are robust (say, optimal) with respect to both n and p. We also note that the W-cycle convergence rate is essentially the same as the corresponding two-grid convergence rate: compare Tables 16–17 with Tables 8 and 14.

8.3. The role of the advection term

In the presented numerical results we considered only the diffusion term of problem (2.1). If we have an advection term, so $\boldsymbol{\beta} \neq \mathbf{0}$, then our discretization matrices are – strictly spoken – not anymore SPD, but they can still be approximately SPD. We briefly investigated the behavior of our algorithms in the presence of a non-zero advection term, and the conclusions are not surprising.

If β_j/n_j is small enough, then the same techniques work unchanged and the iteration count is practically the same as well. Actually, in the 1D case, $\frac{1}{n}A_n^{[p]} = K_n^{[p]} + \frac{\beta}{n}H_n^{[p]} + \frac{\gamma}{n^2}M_n^{[p]}$ can be regarded as $K_n^{[p]}$ plus a matrix whose infinity norm tends to zero as $n \to \infty$, see Lemma 2.1. Therefore, the situation is virtually unchanged, and the same conclusion also holds for every dimensionality d.

Let us now have a look at the case where there exists a $j \in \{1, \ldots, d\}$ such that β_j is proportional to n_j. We note, however, that this is not natural from the viewpoint of the

approximation of equation (2.1). In such a case, the symbol changes completely and so we have to change our algorithms. In particular, a proposal to be investigated, is to maintain the multi-iterative approach by using a P-GMRES method as external solver, preconditioned by a multigrid method of the kind we have considered so far and a specific preconditioner for the non-Hermitian part.

9. Conclusion and perspectives

By following the multi-iterative approach and by using the knowledge of the symbol, we have designed an effective iterative solver for large linear systems arising from the Galerkin B-spline method approximating classical d-dimensional elliptic problems, $d \geq 1$. The main features of the technique are:

1. it has an optimal global cost, i.e., the overall number of operations is proportional to the number of degrees of freedom;
2. it is totally robust, i.e., its convergence speed is substantially independent of all the relevant parameters, namely the matrix size (related to the finesse parameter), the spline degree p (associated to the approximation order), and the dimensionality d of the problem.

Besides several theoretical issues related to the rigorous proofs of optimal and robust convergence of the proposal, the most intriguing challenge is an extension which is able to capture the geometrical mapping (in the case of non-trivial physical domains) and the variable coefficients in a more general elliptic operator of the form:

$$-\nabla \cdot (K \, \nabla \mathbf{u}) + \text{lower order terms},$$

on $\Omega = G([0,1]^d)$, with G a geometric map and $K : \Omega \to \text{Symm}_d$, Symm_d being the set of all real symmetric square matrices of size d. In that direction we see two main future steps:

1. the computation of the symbol and its analysis for determining approximately the critical subspaces of ill-conditioning; it is worth noticing that we expect that the global symbol of the associated matrix sequences can be formed, in analogy with the FD/FE context [4, 35, 36], by using the information from the main operator (the principal symbol in the Hörmander theory [26]), the used approximation techniques, and the involved domain i.e. the geometric map G;
2. the design of an ad hoc multi-iterative strategy (a single basic iteration for each specific critical subspace) with the goal of reaching optimality and total robustness with respect to all the parameters: n (the finesse parameter), p (the approximation order), d (the dimensionality), G (the geometry), K and the advection term (the PDE coefficients).

References

[1] A. Aricò, M. Donatelli. *A V-cycle Multigrid for multilevel matrix algebras: proof of optimality*. Numer. Math. **105** (2007), 511–547.

[2] A. Aricò, M. Donatelli, S. Serra-Capizzano. *V-cycle optimal convergence for certain (multilevel) structured linear systems*. SIAM J. Matrix Anal. Appl. **26** (2004), 186–214.

[3] O. Axelsson. *Iterative solution methods*. Cambridge University Press (1996).

[4] B. Beckermann, S. Serra-Capizzano. *On the asymptotic spectrum of Finite Elements matrices*. SIAM J. Numer. Anal. **45** (2007), 746–769.

[5] L. Beirão da Veiga, D. Cho, L.F. Pavarino, S. Scacchi. *BDDC preconditioners for isogeometric analysis*. Math. Models Methods Appl. Sci. **23** (2013), 1099–1142.

[6] L. Beirão da Veiga, D. Cho, L.F. Pavarino, S. Scacchi. *Isogeometric Schwarz preconditioners for linear elasticity systems*. Comput. Methods Appl. Mech. Engrg. **253** (2013), 439–454.

[7] R. Bhatia. *Matrix analysis*. Springer-Verlag, New York (1997).

[8] C. de Boor. *A practical guide to splines*. Springer-Verlag, New York (2001).

[9] A. Böttcher, S. Grudsky. *On the condition numbers of large semi-definite Toeplitz matrices*. Linear Algebra Appl. **279** (1998), 285–301.

[10] A. Böttcher, S. Grudsky, E. Ramirez de Arellano. *On the asymptotic behavior of the eigenvectors of large banded Toeplitz Matrices*. Mathematische Nachrichten **279** (2006), 121–129.

[11] H. Brezis. *Functional analysis, Sobolev spaces and partial differential equations*. Springer (2011).

[12] A. Buffa, H. Harbrecht, A. Kunoth, G. Sangalli. *BPX-preconditioning for isogeometric analysis*. Comput. Methods Appl. Mech. Engrg. **265** (2013), 63–70.

[13] J.A. Cottrell, T.J.R. Hughes, Y. Bazilevs. *Isogeometric analysis: toward integration of CAD and FEA*. John Wiley & Sons (2009).

[14] F. Di Benedetto, G. Fiorentino, S. Serra. *C.G. preconditioning for Toeplitz matrices*. Comput. Math. Appl. **25** (1993), 33–45.

[15] M. Donatelli. *An algebraic generalization of local Fourier analysis for grid transfer operators in multigrid based on Toeplitz matrices*. Numer. Linear Algebra Appl. **17** (2010), 179–197.

[16] M. Donatelli, C. Garoni, C. Manni, S. Serra-Capizzano, H. Speleers. *Symbol-based multigrid (and multi-iterative) methods for Galerkin B-spline isogeometric analysis*. In preparation.

[17] M. Donatelli, S. Serra-Capizzano. *On the regularizing power of multigrid-type algorithms*. SIAM J. Sci. Comput. **26** (2006), 2053–2076.

[18] M. Donatelli, S. Serra-Capizzano, D. Sesana. *Multigrid methods for Toeplitz linear systems with different size reduction*. BIT Numer. Math. **52** (2012), 305–327.

[19] H. Engl, M. Hanke, A. Neubauer. *Regularization of Inverse Problems*. Kluwer Academic Publishers, Dordrecht, The Netherlands (1996).

[20] G. Fiorentino, S. Serra. *Multigrid methods for Toeplitz matrices*. Calcolo **28** (1991), 283–305.

[21] G. Fiorentino, S. Serra. *Multigrid methods for symmetric positive definite block Toeplitz matrices with nonnegative generating functions*. SIAM J. Sci. Comput. **17** (1996), 1068–1081.

[22] K.P.S. Gahalaut, J.K. Kraus, S.K. Tomar. *Multigrid methods for isogeometric discretization*. Comput. Methods Appl. Mech. Engrg. **253** (2013), 413–425.

[23] K.P.S. Gahalaut, S.K. Tomar, J.K. Kraus. *Algebraic multilevel preconditioning in isogeometric analysis: Construction and numerical studies*. Comput. Methods Appl. Mech. Engrg. **266** (2013), 40–56.

[24] C. Garoni. *Estimates for the minimum eigenvalue and the condition number of Hermitian (block) Toeplitz matrices*. Linear Algebra Appl. **439** (2013), 707–728.

[25] C. Garoni, C. Manni, F. Pelosi, S. Serra-Capizzano, H. Speleers. *On the spectrum of stiffness matrices arising from isogeometric analysis*. Numer. Math. (2013), to appear.

[26] L. Hörmander. *Pseudo-differential operators and non-elliptic boundary problems*. Annals of Math. **2** (1966), 129–209.

[27] R.A. Horn, C.R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press (1994).

[28] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs. *Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*. Comput. Methods Appl. Mech. Engrg. **194** (2005), 4135–4195.

[29] E. Ngondiep, S. Serra-Capizzano, D. Sesana. *Spectral features and asymptotic properties for g-circulants and g-Toeplitz sequences*. SIAM J. Matrix Anal. Appl. **31** (2010), 1663–1687.

[30] S. Serra. *Multi-iterative methods*. Comput. Math. Appl. **26** (1993), 65–87.

[31] S. Serra. *On the extreme spectral properties of symmetric Toeplitz matrices generated by L^1 functions with several global minima/maxima*. BIT **36** (1996), 135–142.

[32] S. Serra. *On the extreme eigenvalues of Hermitian (block) Toeplitz matrices*. Linear Algebra Appl. **270** (1998), 109–129.

[33] S. Serra-Capizzano. *An ergodic theorem for classes of preconditioned matrices*. Linear Algebra Appl. **282** (1998), 161–183.

[34] S. Serra-Capizzano. *Convergence analysis of two-grid methods for elliptic Toeplitz and PDEs matrix-sequences*. Numer. Math. **92** (2002), 433–465.

[35] S. Serra-Capizzano. *Generalized locally Toeplitz sequences: spectral analysis and applications to discretized partial differential equations*. Linear Algebra Appl. **366** (2003), 371–402.

[36] S. Serra-Capizzano. *The GLT class as a generalized Fourier analysis and applications*. Linear Algebra Appl. **419** (2006), 180–233.

[37] S. Serra-Capizzano, C. Tablino-Possio. *Multigrid methods for multilevel circulant matrices*. SIAM J. Sci. Comput. **26** (2004), 55–85.

[38] P. Tilli. *A note on the spectral distribution of Toeplitz matrices*. Linear Multilinear Algebra **45** (1998), 147–159.

[39] W.F. Trench. *Properties of unilevel block circulants.* Linear Algebra Appl. **430** (2009), 2012–2025.

[40] W.F. Trench. *Properties of multilevel block α-circulants.* Linear Algebra Appl. **431** (2009), 1833–1847.

[41] U. Trottenberg, C.W. Oosterlee, A. Schüller. *Multigrid.* Academic Press, London (2001).

[42] N.L. Zamarashkin, E.E. Tyrtyshnikov. *On the distribution of eigenvectors of Toeplitz matrices with weakened requirements on the generating function.* Russian Math. Survey **522** (1997), 1333–1334.

[43] R.S. Varga. *Matrix iterative analysis.* Prentice Hall, Englewood Cliffs, 1962.